

République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université A. MIRA-Bejaia
Faculté de Technologie
Département De Génie Electrique



MEMOIRE DE FIN DE CYCLE

En vue de l'obtention du diplôme de Master en Electrotechnique

Option : Automatismes Industriels

Thème

**Etude d'un système d'exploitation de siroperie
Tetra avec sauvegarde des données**

Réalisé par :

Mr. MESSAHLI Redha

Mr. YAHIAOUI Hicham

Encadré par :

Mr. TAIB Nabil

Mr. RABHI Djamel

Promotion 2017

Remerciements

Tout d'abord, nous tenons à remercier, «Allah» le tout puissant qui nous a procuré, patience, courage et volonté afin de réaliser ce modeste travail.

Au terme de la réalisation de ce travail, nous remercions vivement notre promoteur Mr. TAIB Nabil. On lui suit reconnaissants de nous avoir accepté de nous encadrer, on le remercie pour tous ses conseils, sa disponibilité, son sérieux dans le travail, on est très fier d'avoir eu la chance de travailler sous sa direction.

On souhaite remercier tous les enseignants pour leur disponibilité, gentillesse et leur soutien qui nous ont apporté au cours de nos années d'étude.

On remercie sincèrement les membres du jury qui nous ont fait l'honneur d'accepter de juger ce travail.

Nous présentons aussi nos vifs et sincères remerciements au directeur qualité Mr. RABHI Djamel qui nous à acceptés à effectuer notre stage et de nous encadrer, on le remercie pour tous l'aide qu'il nous a fournie, sa disponibilité, sans sérieux de travail, on est vraiment très fier d'avoir eu la chance de travailler sous sa direction, on le remercie aussi pour leur bien accueillis au sein de leur organisme "SARL IBRAHIM & FILS -IFRI "

Nous aimerions également remercies toutes les personnes avec qui on a eu l'occasion de travailler dans cette organisme, tous ceux qui nous ont guidé, conseillé et surtout orienté à savoir : Mr CHAIT Younes et AMGHAR Arezki, Mr Boudjemaa et Yanis, et tous les contrôleurs et les employés.

Nous remercions tous ceux qui ont contribué de près ou de loin au bon déroulement de ce stage. Nous gardons l'image d'une entreprise organisée, propre, respectueuse.

Ce fut un plaisir et un honneur de travailler avec vous.

Dédicace

C'est avec profonde gratitude et sincères mots, que je dédie ce modeste travail de fin d'étude à

Mes chers parent qui ont sacrifiée leur vie pour ma réussite et mon éclairé le chemin par leurs conseils judicieux.

J'espère qu'un jour, je pourrais leurs rendre un peu de ce qu'ils ont fait pour moi.

Je dédie aussi ce travail à mes grands-parents, à mon chère amie et Co-binôme Hicham pour tous les moments de joies et de peines qu'on a passées ensemble, a sa famille aussi.

Je dédie aussi ce travail à mes deux sœurs, ma grande famille, mes amies, tous mes collègues de la spécialité automatisme et tous nos professeurs qui nous ont enseigné et à tous ceux qui me sont chers.

MESSAHLI Redha

Dédicace

Je dédie le fruit de ce modeste travail comme

Un geste de gratitude à :

La mémoire de mon grand-père Abdelkader

Ma grande mère Hadda

Mes très chers parents, Rachid et Zahoua qui m'ont soutenu,

encouragé pour que je puisse mener à bien mes

études, et qui ont attendu ce jour avec impatience.

Mon frère Yougourtha

Ma Sœur Souad

Mon binôme Redha

*À tous mes amis, il m'est impossible de vous citer tous nommément,
tant que vous êtes nombreux.*

A tous mes collègues du groupe automatisme

*A tout ceux qui ont contribué à la réalisation de
ce travail de loin ou de près.*

Y. Hicham

TABLES DES MATIERES

Table des figures	v
Liste des tableaux	viii
Liste des abréviations	ix
Introduction Général	1
Présentation de la SARL IBRAHIM & fils, IFRI	3
Introduction.....	3
I Analyse fonctionnelle et identification des équipements et instruments	7
Introduction	7
I.1 Service de production (siroperie)	7
I.2 Identification des actionneurs de la siroperie	7
I.2.1 Les pompes.....	7
I.2.1.1 Pompe Centrifuge	7
I.2.1.2 Pompe à anneau liquide	8
I.2.1.2 Pompe Volumétrique	8
I.2.2 Les vannes	9
I.2.2.1 Vanne manuelle	9
I.2.2.2 Vanne modulante	9
I.2.2.3 Vanne de régulation	10
I.3 Identification des instruments de la siroperie	10

I.3.1 Sonde de pression.....	10
I.3.2 Sonde de température.....	10
1.3.3 Capteur de niveau.....	11
I.3.4 Débitmètre.....	11
I.3.5 Indicateur de pression.....	12
I.4 Description des éléments des armoires électrique de process et des machines.....	12
I.4.1 Les disjoncteurs.....	12
I.4.1.1 Disjoncteur principal.....	12
I.4.1.2 Disjoncteur magnétothermique.....	13
I.4.1.3 Disjoncteur magnétique.....	13
I.4.2 Les relais.....	13
I.4.2.1 Relai de commande.....	13
I.4.2.2 Relai de sécurité.....	14
I.4.3 Thermostat.....	14
I.4.4 Le sectionneur.....	14
I.4.5 Variateur de vitesse.....	15
I.4.6 Alimentation AS-I BUS.....	15
I.5 Elaboration de l’analyse fonctionnelle du système de la siroperie.....	16
I.5.1 La Zone 20 (Stockage de sirop).....	16
I.5.2 La Zone 30(ALMIX).....	17
I.5.3 Zone 40 (Concentrait).....	19
I.5.4 Zone 50 (Reconstitution).....	20
I.5.5 Zone 60 Pasteurisateur (DRINK).....	22
I.5.6 La Zone 70(ALSAFE).....	24
I.5.7 La Zone 80(rejet).....	25
I.5.8 Zone 200 (NEP).....	26
I.5.9 Zone10 (de fabrication de sirop).....	27
Contexte et Problématique.....	27
Conclusion.....	28
II. Généralité sur les Automates et logiciel associés et les différentes liaisons	29
Introduction.....	29

II.1 Généralité sur l'Automate Programmable Industriel.....	29
II.1.1 Définition.....	29
II.1.2 Structure interne de l'Automate	29
II.1.3 Critères de choix d'un automate	30
II.1.4 Les avantages de l'API	30
II.1.5 Les inconvénients	30
II.2 AUTOMATE PROGRAMMABLE SIMATIC S7-300.....	30
II.2.1 Définition.....	30
II.2.2 Composants de S7-300.....	31
II.2.3 Les Interfaces.....	31
II.2.3.1 Interface MPI	31
II.2.3.2 Interface Ethernet.....	32
II.2.3.3 Interface Profibus.....	32
II.3 Généralité sur KEPServerEx	32
II.3.1 Définition de KEPServer.....	32
II.3.2 Application client / serveur	33
II.3.2.1 Architecture d'une Application clients/serveurs	33
II.3.3 OPC	33
II.3.4 Optimisation des communications Ethernet TCP / IP Siemens	33
II.4 Généralité sur le système SCADA « InTouch »	34
II.4.1 Le logiciel InTouch	34
II.4.2 InTouch HMI : visualisation du processus de fabrication.....	35
II.4.3 Création d'application	35
II.4.4 Gestion des applications	36
II.4.5 Gestion de données	37
II.4.6 Type de données	37
II.4.7 Propriétés des variables	37
II.4.8 Fenêtres InTouch	38
II.4.9 Points forts du logiciel InTouch.....	38
II.5 Généralité sur les bases de données.....	39
II.5.1 Définition d'une base de données.....	39
II.5.2 Les bases de données relationnelles.....	39

II.5.3. Les systèmes de gestion de bases de données (SGBD)	39
II.5.3.1 Notion d'un SGBD.....	39
II.5.3.2 Objectifs des SGBD	40
II.5.3.3. Les fonctions des SGBD	41
II.5.4 langage SQL	41
II.5.5 Le langage T-SQL	42
II.5.6 Microsoft SQL Server 2008	42
II.6 Les différentes liaisons	43
II.6.1 Liaison Siemens S7-300 avec KEPServer	43
II.6.2 Liaison SCADA InTouch avec KEPServer	45
II.6.3 Liaison InTouch avec SQL Server 2008.....	46
II.7 Association de variables InTouch aux colonnes de la base de données	46
Conclusion.....	48
II. Exemple de sauvegarde et teste de requêtes	49
Introduction	49
III.1 Identification des paramètres de la zone 60 (DRINK) à sauvegarder sur la base de données	49
III.2 Langage de scripts InTouch.....	50
III.3 Fonctions SQL utilisé dans le Script	50
III.3.1.1 Fonction SQLConnect	50
III.3.1.2 Fonction SQLDisconnect	50
III.3.1.3 Fonction SQLExecute	50
III.3.1.4 Fonction SQLErrorMsg	50
III.3.1.5 Fonction SQLNext.....	50
III.3.1.6 Fonction SQLPrepareStatement	51
III.3.1.7 Fonction SQLSetStatement	51
III.3.1.8 Fonction SQLAppendStatement	51
III.3.1.9 Fonction SQLClearStatement.....	51

III.3.1.10 Fonction StringFromReal	51
III.4 Création d'une base de données	52
III.5 Création des tables	53
III.6 Algorithme du script utilisé sur InTouch pour la Sauvegarde de données dans tetra154	
III.7 Algorithme utilisée dans Sage 100 Edition Pilotée à fin de récupérer les valeurs de la base de données tetra1	55
III.8 Test requête SQL Server.....	55
Conclusion.....	57
Conclusion générale	58
Bibliographie	59
ANEXE	60

Table des figures

Figure1 : Logo de la SARL «IFRI »	3
Figure2 : Vue depuis le satellite d'IFRUIT by IFRI.....	5
Figure3 : Organigramme de la SARL IBRAHIM et fils, IFRI	6
FigureI.1 : Pompe centrifuge	8
FigureI.2 : Pompe à anneau liquide	8
FigureI.3 : Pompe Volumétrique	8
FigureI.4 : Vanne avec robinet	9
FigureI.5 : Vanne modulante SAMSON.....	9
FigureI.6 : Vanne régulatrice	10
FigureI.7 : Sonde de pression Tetra Pack	10
FigureI.8 : Sonde de température Tetra Pack.....	11
FigureI.9 : Capteur de niveau.....	11
FigureI.10 : Débitmètre.....	11
FigureI.11 : Indicateur de pression	12
FigureI.12 : disjoncteur principal Schneider.....	12
FigureI.13 : Disjoncteur magnétothermique	13
FigureI.14 : disjoncteur thermique.....	13
FigureI.15 : Relai de commande	13
FigureI.16 : Relai de sécurité	14

FigureI.17 : Thermostat	14
FigureI.18 : Sectionneur.....	14
FigureI.19 : Variateur de vitesse Danfoss.....	15
FigureI.20 : alimentation AS-I BUS	15
FigureI.21 : Vue de l'unité Stockage de Sirop.....	16
FigureI.22 : Vue de l'unité ALMIX.....	17
FigureI.23 : Vue de l'unité Concentrait	19
FigureI.24 : Vue de l'unité Reconstitution	20
FigureI.25 : Vue de l'unité Pasteurisateur	22
FigureI.26 : Vue de l'unité ALSAFE.....	24
FigureI.27 : Vue de l'unité Rejet	25
FigureI.28: Vue de l'unité NEP	26
FigureII.1 : Schéma explicatif.....	29
FigureII.2 : Structure d'un API.....	30
FigureII.3 : Vue externe de l'Automate S7-300	31
FigureII.4 : Vue général de l'Automate S7-300	31
FigureII.5 : Exemple d'une configuration à l'aide d'un seul canal.....	34
FigureII.6 : Exemple d'une configuration à l'aide de plusieurs canaux.....	34
Figure II.7 : Les outils de WindowMaker permettant de créer des applications HMI.	36
Figure II.8 : Démonstration comment les applications InTouch sont créées puis administrées avec le Gestionnaire d'applications.....	36
FigureII.9 : la boîte de dialogue du Dictionnaire de variables	37
FigureII.10: les fenêtres de WindowMaker	38
FigureII.11 : Modèle de données relationnel	39
Figure II.12 : Capture de fenêtre de création d'un nouveau Channel	43
FigureII.13 : Capture de sélection de pilote de périphérique.....	43
FigureII.14 : vue du choix du périphérique	44
FigureII.15 : Capture qui décrit toute les informations du Channel créé.....	44
FigureII.16: Capture de modification de nom d'accès.....	45
FigureII.17: vue d'une fenêtre création d'une de carte d'alias	45
FigureII.19 : Capture d'une fenêtre InTouch	46

FigureIII.3 : Capture représentant une liste de liens associe les colonnes de la base de données aux variables InTouch 47

FigureIII.1 : Interface Microsoft SQL Server Management Studio..... 52

Figure III.2 : Fenêtre explorateur représentant les deux tables TETRA_DR1 et TETRA_DR2créé avec les données des colonnes..... 53

Figure III.3 : Algorithme du script de sauvegarde de données sur InTouch dans la base tetra1 54

FigureIII.4 : Algorithme du script de récupération des données de la table TETRA_DR1. 55

Figure III.5 : Courbe de données de trois variables d'une requête formulée..... 56

Figure III.6 : Courbe de données de deux variables d'une requête formulée. 57

Figure III.7 : Courbe de données de deux variables d'une requête formulée..... 57

Liste des tableaux

Tableau III.1 : variables de la zone 60 (DRINK) à sauvegarder.	49
TableauxIII.2 : Tableau des données sauvegardées sur la table 1 de la base tetra1.	55
Tableau III.3 : Tableau des données sauvegardées sur la table 2 de la base tetra1.	56

Liste des Abréviations

PET : Polytéraphalate d'éthylène.

SARL : Société à Responsabilité Limité.

SQL : Structured Query Language.

SPC : Statistical Process Control.

NEP : Nettoyage En Place.

DC : Direct courant.

API : Automate Programmable Industriel.

IES : Interface Entrée Sortie.

RAM : Random Access Memory.

EEPROM : Electrically-Erasable Programmable Read-Only Memory.

TOR : Tout Ou Rien.

EPROM : Erasable Programmable Read-Only Memory.

OPC : OLE for Process Control.

IP : Internet Protocol.

TCP : Transmission Control Protocol.

BD : Base de Données.

Introduction Général

Dans le monde industriel, les exigences attendues de l'automatisation ont bien évolué avec la progression continue de la technologie. Les demandes et les exigences actuelles des entreprises ne s'arrêtent pas uniquement à l'augmentation de la productivité, ou la diminution des coûts de production, mais elles concernent aussi l'amélioration des conditions de travail, l'élimination des tâches pénibles, et fatigantes.

L'ingénierie et le développement logiciel, améliorées par les méthodes et processus de développement, sont devenues des concepts fondamentaux à la réussite des entreprises, qui ont trouvé dans l'impossibilité d'avancer sans de bons moyens de gestion de données. En effet, les applications ont réduit les pertes et ont grandement contribué à l'évolution de ces dernières.

Le service de production siroperie de l'entreprise IFRI, se retrouve confronté au problème d'absence de traçabilité, ce qui constitue une charge importante sur l'entreprise.

Le travail qui nous a été confié consiste à faire une étude du système d'exploitation de la siroperie et identifier les paramètres nécessaires au bon fonctionnement du système et qui répondent aux normes du produit, cela est dans le but de sauvegarder ces variables et arriver à identifier d'une manière directe la source des problèmes, cela dans le cas d'une réclamation client ou d'un défaut du produit pour faire avancer la productivité et améliorer les conditions de travail.

Le présent document expose les différentes étapes franchies afin de mener à bien ce projet qui se traduit à travers trois chapitres.

Dans le premier chapitre nous allons parler du service de siroperie et identifier les différents équipements utilisés (actionneurs et instruments) ainsi que les éléments des armoires, ensuite élaborer une analyse fonctionnelle pour bien comprendre le fonctionnement des différentes zones existantes dans la siroperie.

Dans le deuxième chapitre nous allons voir les automates programmables en général, en particulier l'automate S7-300, ensuite nous allons présenter et définir le système de communication « KEPServer », ainsi que le système de supervision SCADA « InTouch » et

puis les bases de données et leur système de gestion. Ensuite nous expliquerons comment créer des liaisons entre Siemens avec KEPServer, KEPServer avec InTouch et InTouch avec Microsoft SQL Server 2008.

Dans le troisième chapitre nous allons récupérer les données des paramètres de production du système et faire des tests de requête à l'aide de l'application Sage 100 Edition Pilotée et visualisée ces donnée.

Enfinement nous terminerons notre travail avec une conclusion Générale.

Présentation de la SARL IBRAHIM et fils, IFRI

Introduction

IFRI est une marque algérienne d'eau minérale et de boissons diverses (sodas et eaux fruitées) appartenant au Groupe Ibrahim et fils et créée en 1996 à Ouzellaguen en Kabylie. Elle est le leader du marché de l'eau minérale et boissons non alcoolisés en Algérie.

Sa source provient du village d'IFRI (où s'est déroulé le congrès de la Soummam le 20 août 1956).

Dans ce présent chapitre nous allons décrire son évolution historique, ses multiples activités industrielles, ses principaux objectifs, ainsi un organigramme décrivant ses différentes directions et services.

Présentation de la SARL IBRAHIM & fils, IFRI

Historique

Le Groupe Ibrahim et fils est pionnier dans l'utilisation (PET) en Algérie, inaugure son premier atelier d'embouteillage le 20 juillet 1996. À cette date, plus de 20 millions de bouteilles ont été commercialisées dans tout le pays. Ce chiffre a atteint 48 millions d'unités en 1999 puis 252 millions de litres en 2004 avant de franchir le cap des 500 millions de litres pour les deux emballages PET et verre en 2009. En 2016, près d'un milliard de bouteilles (eaux minérales naturelles et boisson sucrées) ont été mises sur le marché par le groupe IFRI.

Après avoir couvert les besoins du marché national, IFRI se lance dans la conquête du marché international, en exportant désormais ces produits vers : la France, l'Angleterre, l'Espagne, l'Italie, l'Allemagne, la Belgique, le Luxembourg, le Canada, la Tunisie, la Lybie, le Soudan, le Mali, le Niger et les Émirats arabes unis.

En 2009, IFRI a investi dans une toute nouvelle usine dotée d'équipements de production aseptique. Une technologie de pointe. Bâtie sur un terrain de 20HA, l'usine IFRI est dotée des dernières technologies dans le domaine des boissons, totalement équipée par les géants Krones (Allemagne), Sidel et Husky (Canada) référence mondiale dans l'injection PET.

Depuis cinq(5) ans, IFRI a opéré un changement radical dans son mode de fabrication en bannissant notamment tout additif artificiel dans ses processus pour permettre la production de boissons sans conservateurs, sans arômes ni colorants artificiel.

De prime abord, pour la préparation de ses boissons, IFRI veille à la naturalité des ingrédients utilisés et opte pour l'utilisation d'arômes et colorants naturels à partir de matières végétales et donc non artificielles.



Figure1 : Logo de la SARL «IFRI»

Mission et objectifs

Un important investissement dans de nouvelles lignes aseptiques de toute dernière génération, permettant la production de boissons sans conservateurs, a été mis en branle, un concept unique en Algérie. IFRI réussit ainsi à produire des boissons sans conservateurs, sans arômes ni colorants artificiels, la première et la seule unité au Maghreb !

Des lignes de production aseptiques, les plus perfectionnées au monde et les toutes premières installées sur le continent africain, à haute cadence de conditionnement de boissons. Les boissons haut gamme (tels que les jus 100%, les boissons fuitées, les Boissons fruitées lactées, les boissons pour le sport et les sodas) produites sur ses lignes aseptiques, préviennent ainsi l'apparition de certaines pathologies liées à l'ingestion des additifs artificiels (colorants artificiels, conservateur.....etc.).

L'usine d'Akbou dispose de 3 lignes aseptiques entièrement automatisées produisant 500 millions d'unités annuellement avec une moyenne de 2 millions de bouteilles/jours. Le procédé aseptique évite toute contamination via l'air, l'emballage ou le matériel par des micro-organismes d'altération (levures et moisissures) ou des pathogènes.

Le conditionnement aseptique consiste à remplir un produit pasteurisé dans un emballage stérile et sous environnement stérile. La fermeture stérile et étanche de la bouteille évite la contamination du produit stérile et ne nécessite pas un traitement thermique après embouteillage.

Le remplissage aseptique permet de conditionnement à froid de bouteilles en PET et garantit ainsi une sécurité optimale, le respect totale de la qualité des produits sensibles à durée de vie étendue même lorsqu'ils sont stockés à température ambiante, et enfin la préservation des qualités organoleptiques des boissons.

Qualité et avantage

Dans un souci de qualité, IFRI dispose d'un laboratoire d'autocontrôle au niveau de chaque site de production qui est équipés, chacun, d'un matériel ultra-moderne et performant pour la réalisation des analyses nécessaires. A l'aide de sa propre équipe d'ingénieur et technicien contrôle la qualité, elle s'assure au quotidien de la parfaite conformité physico-chimiques, microbiologiques et organoleptiques de ses produits, depuis l'entrée (contrôle des matières premières et des emballages) jusqu'à la sortie (contrôle des produit finis) et pendant toutes les phases de préparation, de conditionnement et stockage.

Pour garantir la qualité de ses produits et éviter toute contamination par les micro-organismes, aucun contact humain n'est autorisé sur les lignes de production. Dans les 3 salles blanches dont dispose l'usine, les préformes sont introduites, puis stérilisées, et soufflées pour prendre les formes des différentes bouteilles des produit IFRI.

Le process ASEPTIQUE permet entre autre conservation longue durée (plus de 6 mois) à température ambiante sans ajout de conservateurs, ainsi les qualités organoleptiques de la boisson sont préservées grâce au remplissage à froid.

Situation géographique

IFRI est implanté zone Industrielle Ahrik, Ighzer, à 50 km à l'ouest de la ville de Bejaïa et à 150 km à l'est d'Alger dans la région de la Petite Kabylie. Ainsi un autre complexe installé à la zone industrielle Taharacht (Akbou).



Figure2 : Vue depuis le satellite d'Ifruit by IFRI.

Organigramme de l'entreprise IFRI

Les différentes direction et service de la SARL IBRAHIM et fils-IFRI sont présentés par la figure suivante :

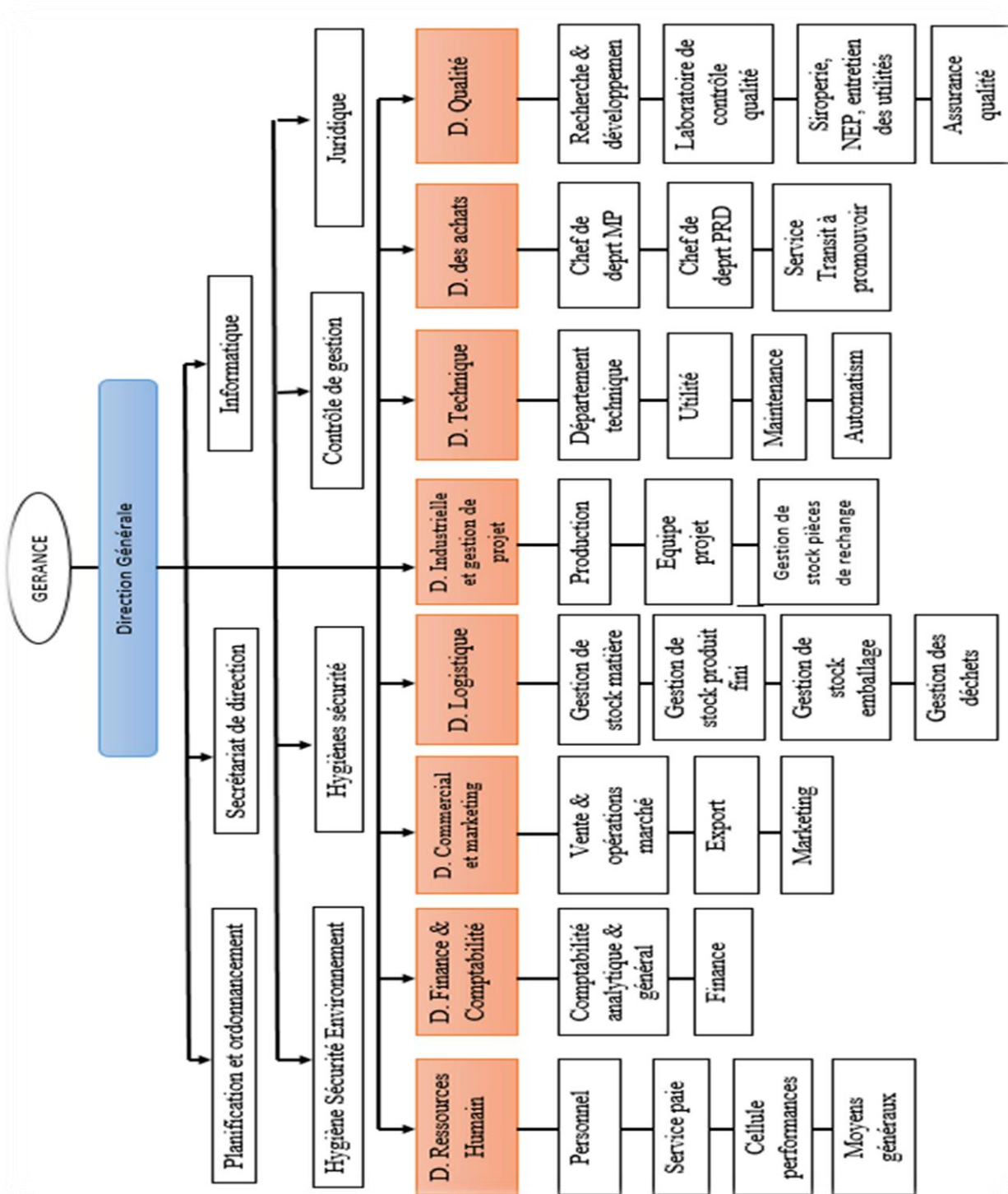


Figure3 : Organigramme de la SARL IBRAHIM et fils, IFRI.

Chapitre I

Analyse fonctionnelle et identification des équipements et instruments

Introduction

L'objectif de ce premier chapitre consiste à présenter le process siroperie et les différents actionneurs et les éléments du système ainsi élaborer une analyse fonctionnelle du système de la siroperie.

I.1 Service de production (siroperie)

Le système de production de la siroperie de la SARL « IFRI » comporte neuf zones de production, chacune d'elles transforme un ensemble de produits déterminés.

La production ce fait en une succession d'opération ordonnées transformant la matière première en un produit fini. Avant de passer par la remplisseuse, l'emballage doit être préparé. S'il s'agit d'emballage en verre, un nettoyage des bouteilles s'impose, les bouteilles ainsi prêtes passe par la remplisseuse puis par une bouchonneuse, ensuite par une étiqueteuse puis par un dateur. Enfin les produits sont mis dans des caisses s'il s'agit d'emballage en verre, ou sou forme de fardeaux si l'emballage et en PET. Les caisses ou les fardeaux sont rassemblés en palettes et seront dirigées vers les stocks.

Ce système de gestion de la siroperie est composé de neuf zones à savoir:

- Zone 20 : Stockage sirop simple.
- Zone 30 : ALMIX.
- Zone 40 : Concentré.
- Zone 50 : Reconstitution.
- Zone 60 : Pasteurisateur (DRINK).
- Zone 70 : ALSAFE (tank aseptique).
- Zone 80 : Incorporation rejet.
- Zone 200 : La station NEP.
- Zone 10 : fabrication de sirop.

I.2 Identification des actionneurs de la siroperie

Au niveau de la siroperie, il existe plusieurs types d'actionneurs. Selon l'utilisation, on distingue les éléments suivants :

I.2.1 Les pompe

Une pompe est un dispositif permettant d'aspirer et de refouler un fluide, et parmi les pompes utilisées on cite :

I.2.1.1 Pompe Centrifuge

La pompe centrifuge peut être utilisée pour pomper tous les liquides de viscosité relativement faible, n'exigeant pas d'être traités avec une douceur particulière. On peut également l'utiliser pour des liquides contenant des particules relativement grosses, pour autant, bien sûr, que la taille de ces particules ne dépasse pas les dimensions des conduits de la turbine.



FigureI.1 : Pompe centrifuge.

I.2.1.2 Pompe à anneau liquide

On utilise les pompes à anneau liquide lorsque le produit contient de grandes quantités d'air ou de gaz et qu'il est donc impossible d'utiliser des pompes centrifuges.

Il en existe une utilisation caractéristique en NEP, comme pompe de retour de la solution de nettoyage en sortie de cuve, la solution de NEP contenant habituellement de grandes quantités d'air.



FigureI.2: Pompe à anneau liquide.

I.2.1.2 Pompe Volumétrique

Une pompe volumétrique est une pompe dans laquelle une certaine quantité de fluide « emprisonnée » est forcée à se déplacer jusqu'à l'orifice de sortie.



FigureI.3 : Pompe Volumétrique.

I.2.1 Les vannes

Comme n'importe quel actionneur elle agit sur une grandeur qui sera toujours pour une vanne de deux voies, la grandeur réglée sera une pression, un débit, un niveau, une température ou un rapport de concentration [8].

I.2.2.1 Vanne manuelle

Ce type de vannes est caractérisé par l'absence du servomoteur sur leur corps, ce qui prive la vanne de la rotation automatique. C'est un levier qui prend place, fonctionnant selon la force qui lui appliquée, permettant ainsi d'voir les positions de fermeture et d'ouverture.



FigureI.4 : Vanne avec robinet.

1.2.2.2 Vanne modulante

Ces vannes automatisées peuvent être utilisées avec des servomoteurs électriques, pneumatiques, électropneumatiques et électrohydrauliques. Les servomoteurs permettent les vannes à être utilisées automatiquement au moyen d'un système de contrôle, soit manuellement à distance. L'automatisation des vannes apporte des avantages significatifs à l'entreprise dans les domaines de la qualité des processus, l'efficacité, la sécurité et la productivité.

Les vannes de régulation peuvent moduler, dans un système, le débit des fluides, la pression, le niveau, et la température.



FigureI.5 : Vanne modulante SAMSON.

1.2.2.3 Vanne de régulation

Est un organe qui relève la régulation industrielle des procédés physico-chimiques. Elle est commandée par un actionneur dont les variations continues de la position modifient la taille de l'orifice de passage du fluide. De cette façon, la chute de pression aux bornes de la vanne est modulée lors du passage d'un fluide, avec pour conséquence la maîtrise du débit traversant.



FigureI.6 : Vanne régulatrice.

I.3 Identification des instruments de la siroperie

La siroperie est équipée d'un ensemble d'instruments qui servent à mesurer les différentes grandeurs : débit, pression, température et niveau. Parmi ces instruments on distingue :

I.3.1 Sonde de pression

Une sonde de pression (ou capteur de pression) est un dispositif destiné à convertir les variations de pression en variations de tension électrique.

Lorsque la sonde est reliée à un système numérique, les variations analogiques sont d'abord converties en signaux numériques binaires par un convertisseur analogique-numérique avant d'être transmises à l'ordinateur de contrôle et de gestion.



FigureI.7 : Sonde de pression Tetra Pack.

I.3.2 Sonde de température

Les sondes de température (ou capteurs de température) sont des dispositifs permettant de transformer l'effet du réchauffement ou du refroidissement sur leurs

composants en signal électrique. Ils peuvent agrandir les signaux de température et puis les transmettre aux signaux DC standards.



FigureI.8 : Sonde de température Tetra Pack.

1.3.3 Capteur de niveau

C'est une sonde de niveau qui est utilisé pour la mesure de niveau dans des réservoirs, c'est-à-dire que le capteur et son conditionneur délivrent un signal proportionnel au niveau de liquide dans le réservoir. À chaque instant, l'opérateur connaît exactement le volume du liquide (ou le volume encore disponible dans le réservoir) [9].



FigureI.9 : Capteur de niveau.

1.3.4 Débitmètre

Un débitmètre est un appareil destiné à mesurer le débit d'un fluide, liquide ou gazeux.



FigureI.10 : Débitmètre.

I.3.5 Indicateur de pression

Dans les processus industriels il est possible d'utiliser les indicateurs de pression. Une surpression, une dépression ou le vide se contrôlent avec les indicateurs de pression. En plus de la simple indication de la pression actuelle, les indicateurs de pression peuvent aussi activer des relais en dépassant les valeurs limites.



FigureI.11 : Indicateur de pression.

I.4 Description des éléments des armoires électrique de process et des machines

I.4.1 Les disjoncteurs

Un disjoncteur, un appareil électromécanique de connexion qui permet de couper le courant d'un ou plusieurs circuits en cas de surintensité ou surcharge, a pour but de protéger :

- Les installations électriques, des courants d'intensités trop élevées susceptibles de les endommager.
- Et l'être humain, de l'électrocution.

Dans les armoires du process et les machines on trouve :

I.4.1.1 Disjoncteur principal

Il est utilisé dans l'armoire du process dans le but de Protection, sectionnement et commande des circuits de distribution électrique. Et il a la possibilité mesure et communication intégrée.



FigureI.12 : disjoncteur principal Schneider.

I.4.1.2 Disjoncteur magnétothermique

Il est utilisé pour assurer la protection contre le court-circuit, les surcharges et la protection des personnes contre les contacts indirects.



FigureI.13 : Disjoncteur magnétothermique.

I.4.1.3 Disjoncteur magnétique

Il est utilisé pour assurer la protection contre le court-circuit.



FigureI.14 : disjoncteur thermique.

I.4.2 Les relais

Un relais est un organe électrique permettant de dissocier la partie puissance de la partie commande il permet l'ouverture et la fermeture d'un circuit électrique par un second circuit complètement isolé (isolation galvanique) et pouvant avoir des propriétés différentes.

I.4.2.1 Relai de commande

Un relais (communément appelé contacteur de puissance*) est composé principalement d'un électroaimant, qui lorsqu'il est alimenté, transmet une force à un système de commutation électrique : les contacts.



FigureI.15 : Relai de commande.

I.4.2.2 Relai de sécurité

Les relais de sécurité contrôlent et surveillent un système de sécurité ; ils autorisent le démarrage de la machine ou exécutent des commandes pour l'arrêter.



FigureI.16 : Relai de sécurité.

I.4.3 Thermostat

Un thermostat est un dispositif permettant de maintenir un système (appareil, machine, moteur, etc.) à une température relativement stable



FigureI.17 : Thermostat.

I.4.4 Le sectionneur

Le sectionneur est un appareil électromécanique permettant de séparer, de façon mécanique, un circuit électrique et son alimentation, tout en assurant physiquement une distance de sectionnement satisfaisante électriquement.



FigureI.18 : Sectionneur.

I.4.5 Variateur de vitesse

Un variateur de vitesse est un équipement permettant de faire varier la vitesse d'un moteur, une nécessité pour de nombreux procédés industriels



FigureI.19 : Variateur de vitesse Danfoss.

I.4.6 Alimentation AS-I BUS

La liaison physique est composée d'une simple paire de fils gainés d'un isolant électrique jaune (pour la commande) ou noir (pour la puissance). Sur ces fils transitent les données ainsi que l'alimentation électrique.

Le bus AS-I étant un bus série, il permet de diminuer le nombre de câbles nécessaire à l'utilisation d'un grand nombre de capteurs ou de pré-actionneurs : ces derniers sont reliés directement ou par l'intermédiaire d'une embase à l'automate central via un seul câble. Chaque embase permet de connecter jusqu'à 4 capteurs ou pré-actionneurs au bus.



FigureI.20 : alimentation AS-I BUS.

I.5 Elaboration de l'analyse fonctionnelle du système de la siroperie

I.5.1 La Zone 20 (Stockage de sirop)

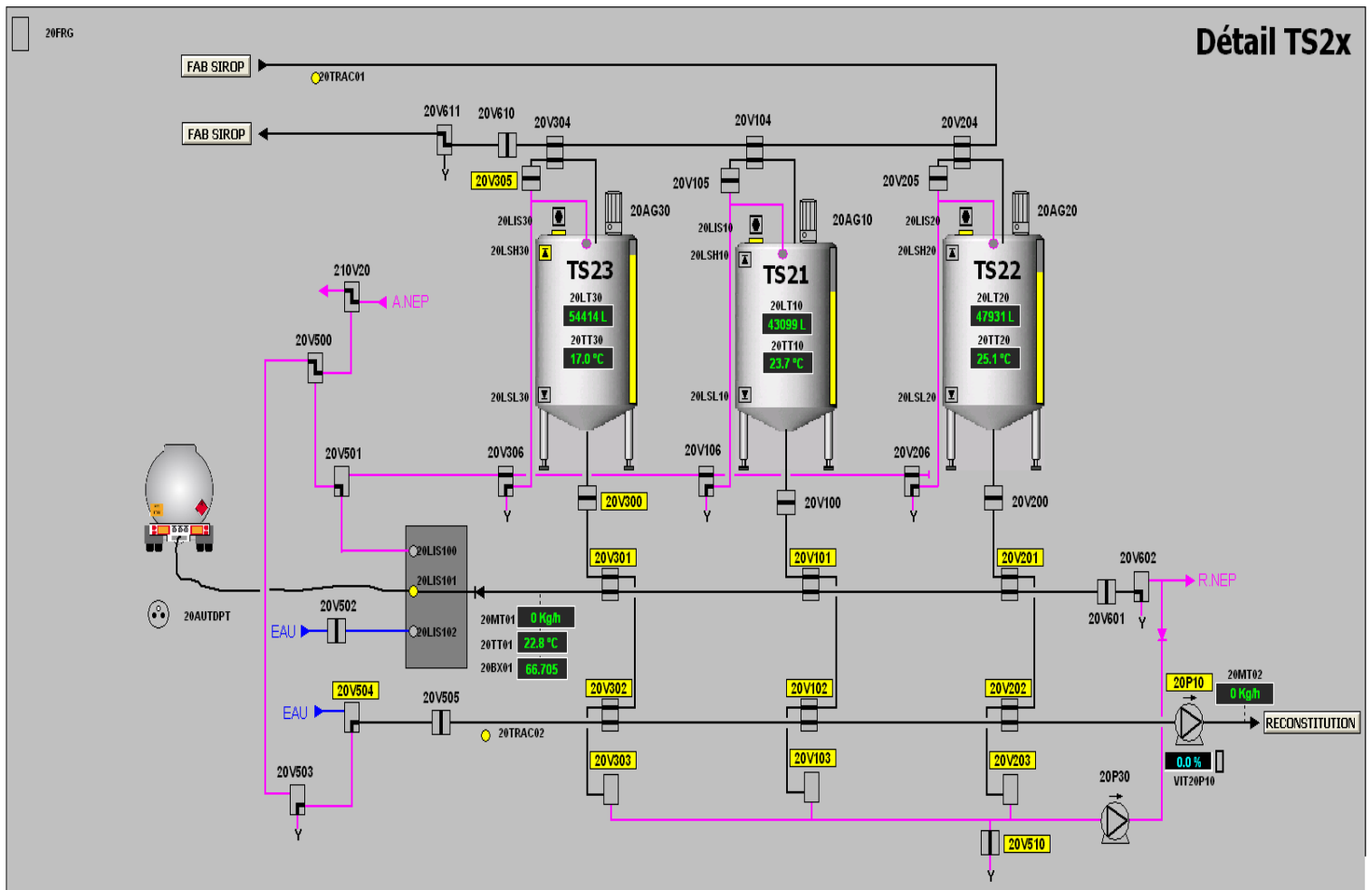


Figure I.21 : Vue de l'unité Stockage de Sirop.

Cette zone regroupe :

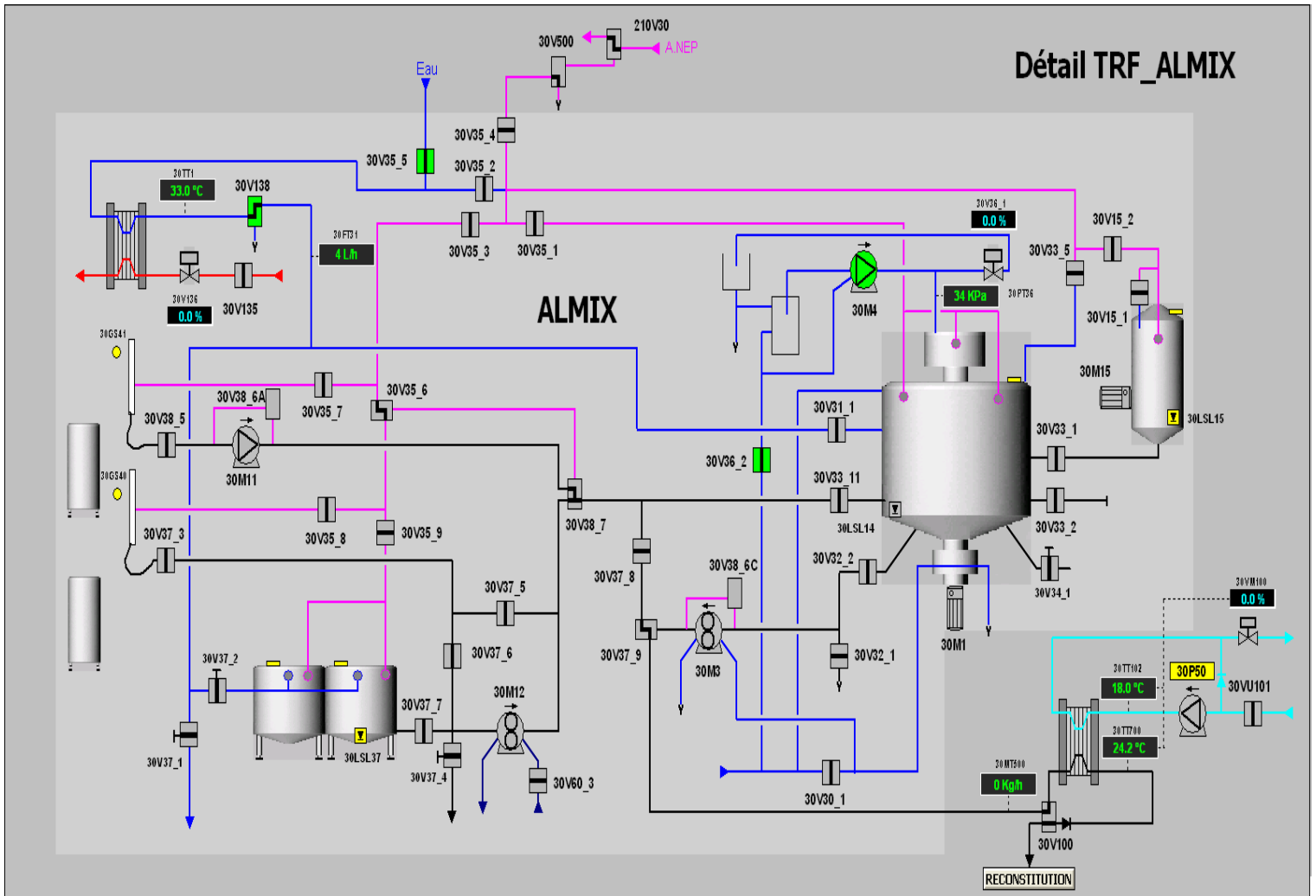
- Trois Tanks (TRS21, TRS22, TRS23).
- Trois capteurs de niveaux bas (20LSL10, 20LSL20, 20LSL30) et trois niveaux hauts (20LSH10, 20LSH20, 20LSH30).
- Trois agitateurs (20AG10, 20AG20, 20AG30) sur chaque tank.
- Trente-quatre vannes (20V100, 20V101, 20V102, 20V103, 20V104, 20V105, 20V106, 20V200, 20V201, 20V202, 20V203, 20V204, 20V205, 20V206, 20V300, 20V301, 20V302, 20V303, 20V304, 20V305, 20V306, 20V500, 20V501, 20V502, 20V503, 20V504, 20V505, 20V601, 20V602, 20V610, 20V611, 210V20).
- Quatre sondes de température (20TT01, 20TT10, 20TT20, 20TT30)
- Deux Débitmètres (20MT01, 20MT02).

- Brixmètre (20BX01).
- Deux pompes d'envoi (20P10, 20P30).

Description du fonctionnement

On a trois cuves de stockage du sucre liquide et on distingue deux plateformes (dépotage et soutirage), après branchement du tuileau au camion le capteur dans le tableau de signalisation s'allume qui signifie que le camion est brancher et aussi prêt pour le dépotage, ensuite en sélectionne la cuve destinataire qui peut stocké le contenu du camion ainsi une cuve de sécurité qui s'active lorsque le capteur du niveaux haut de la cuve destinataire est allumé donc les vannes de cette dernière se ferme automatiquement pour que les vanne de la cuve de sécurité puissent s'ouvrir pour stocké le reste du camion. Une fois le camion est vidé et débranché on passe à la pousse avec l'eau en envoyant une quantité d'eau qui est calculé par rapport au volume du tuileau pour dégager tout le sirop qui reste dans ce dernier.

I.5.2 La Zone 30(ALMIX)



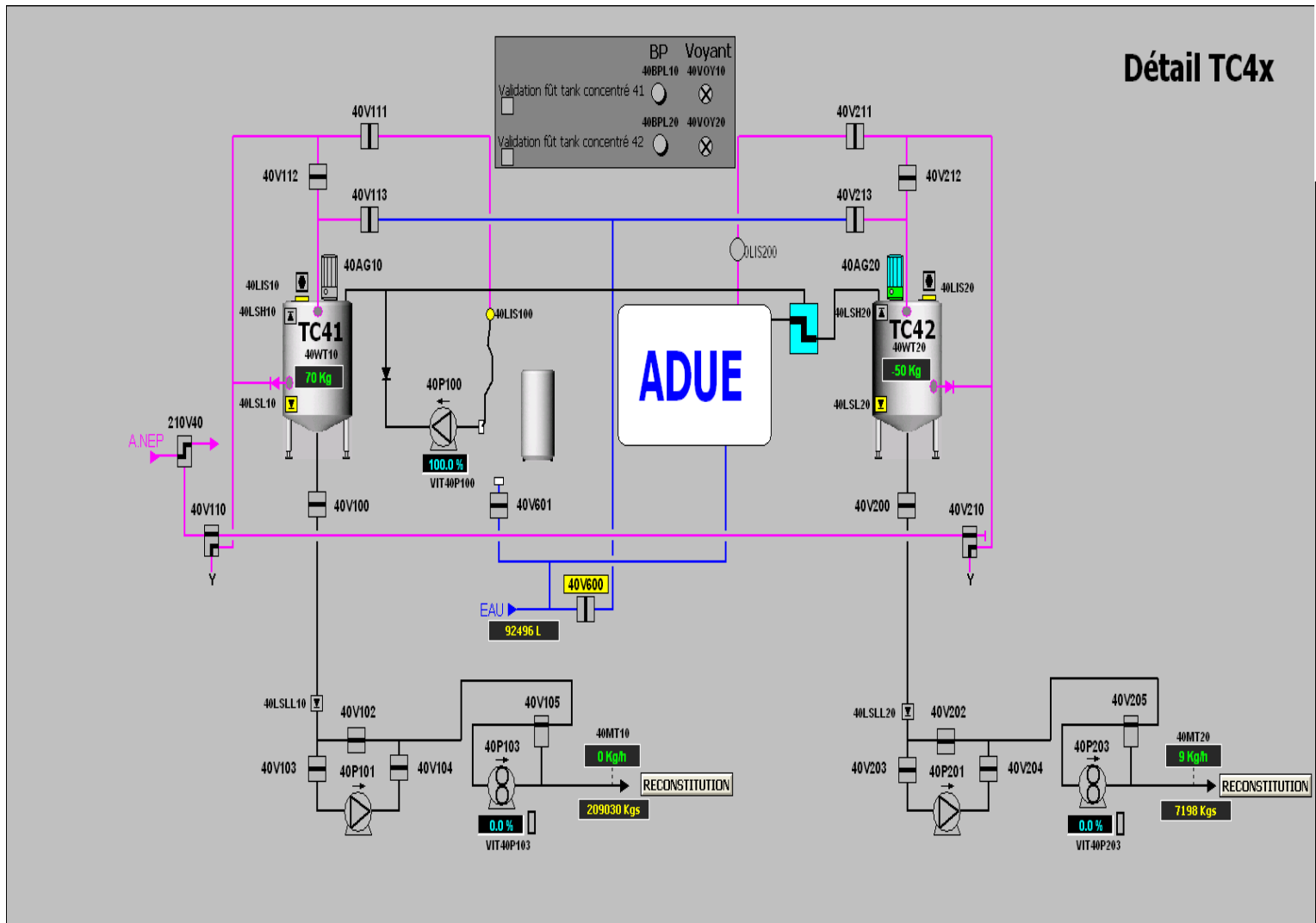
Cette zone regroupe :

- Quarante-quatre vannes, dans trois modulantes (30VM100, 30VM136, 30VM36_1), quatre manuelle (30V34_1, 30V37_1, 30V37_2, 30V37_4) et (30V15_1, 30V15_2, 30V30_1, 30V31_1, 30V32_1, 30V32_2, 30V33_1, 30V33_2, 30V33_11, 30V33_5, 30V34_1, 30V35_1, 30V35_2, 30V35_3, 30V35_4, 30V35_5, 30V35_6, 30V35_7, 30V35_8, 30V35_9, 30V36_2, 30V37_3, 30V37_5, 30V37_6, 30V37_8, 30V37_9, 30V38_5, 30V38_6A, 30V38_6C, 30V38_7, 30V60_3, 30V100, 30V135, 30V138, 30V500, 30VU101, 210V30).
- Deux échangeurs à plaque (un autre de chauffage et un de refroidissement).
- Cinq pompes (30P50, 30M3, 30M4, 30M11, 30M12).
- Deux agitateurs (30M1, 30M15).
- Trois sondes de température (30TT700, 30TT1, 30TT02).
- Deux débitmètres (30FT31, 30MT500).

Description du fonctionnement

On n'a l'eau qui passe par un échangeur de chaleur jusqu'à ce qu'il atteigne la température conditionner, la vanne s'ouvre automatiquement pour laisser passer l'eau à la cuve jusqu'à un niveau voulu puis elle se ferme et on commence à ajouter les ingrédients automatiquement on choisissant une recette, ensuite l'agitateur démarre et il continue à agiter pendant une durée donnée quand il termine l'agitation le produit est prêt pour le transfert et il sort grâce à une pompe et il passe par un refroidisseur. Une fois le produit est sorti on passe à la pousse avec l'eau pour bien nettoyer la citerne et les conduites d'où est passé le produit pour qu'on puisse préparer un autre produit.

I.5.3 Zone 40 (Concentrait)



FigureI.23 : Vue de l'unité Concentrait.

Cette zone regroupe :

- Deux Tanks (TC41, TC42) qui contient le concentré de fruits.
- Deux agitateurs (40AG10, 40AG20) sur chaque tank.
- Deux capteurs de niveaux bas (40LSL10, 40LSL20) et deux niveaux hauts (40LSH10, 40LSH20).
- Vingt-et-un vanne (40V100, 40V102, 40V103, 40V104, 40V105, 40V110, 40V111, 40V112, 40V113, 210V40, 40V600, 40V601, 40V200, 40V202, 40V203, 40V204, 40V205, 40V210, 40V211, 40V212, 40V213).
- Dépotage manuel (40LIS100) ou automatique (ADUE).
- Cinq pompes d'envoi (40P100, 40P101, 40P103, 40P201, 40P203).

- Débitmètre (40MT10, 40MT20).

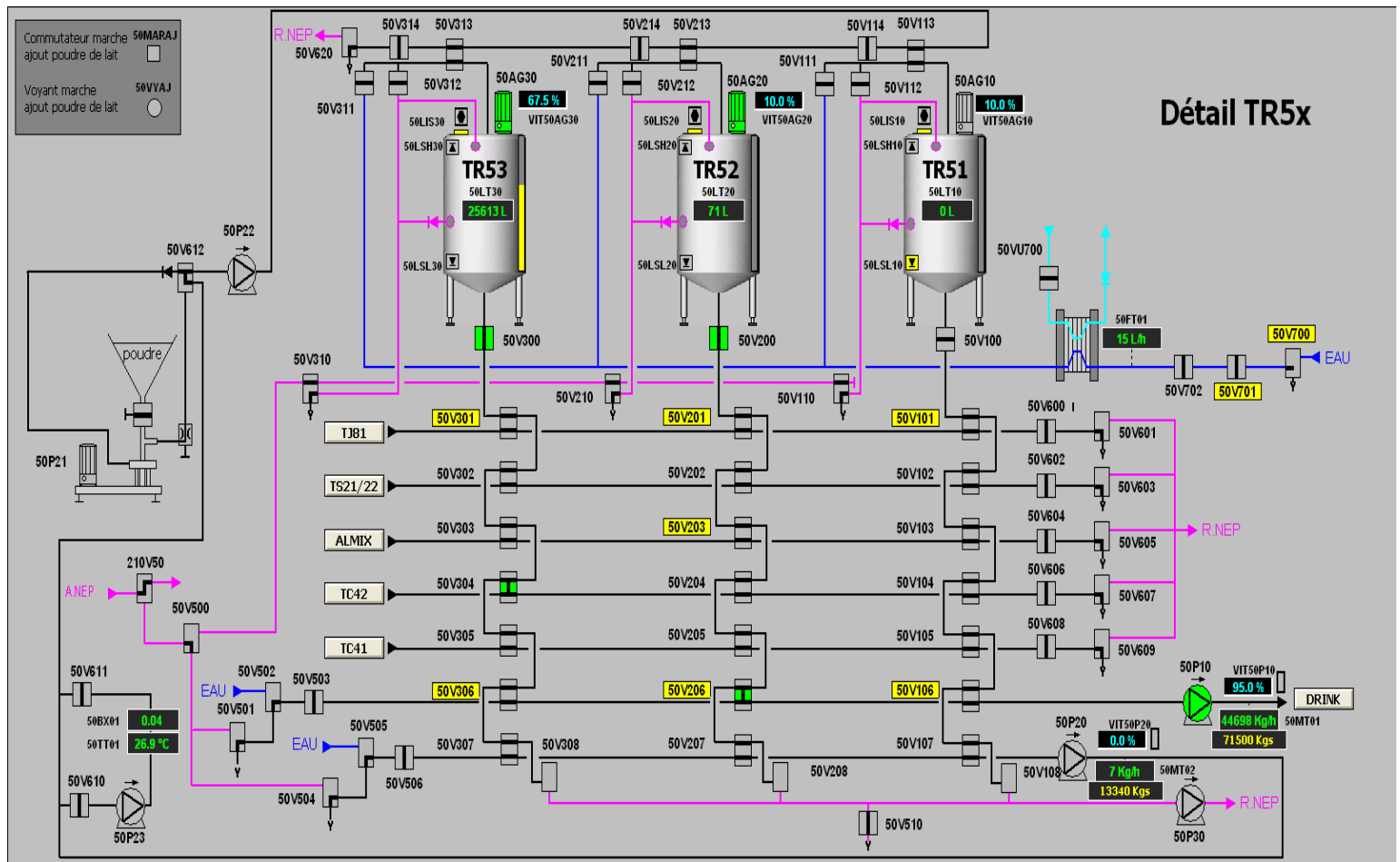
Description du fonctionnement

La zone 40 est composée deux tank pour stockage du concentré de fruits, ces tanks sont muni d’un agitateur, d’une arrivée CIP, arrivée du concentrat, arrivée d’eau, et d’une sortie vers la Zone 50.

Le concentré arrive dans les tanks soit par dépotage manuel ou automatique, grâce à des pompes qui transfèrent le concentré des fûts vers l’un des deux tanks. Selon le tank sélectionné et grâce à une pompe d’envoi le concentré est envoyé à la zone 50.

Après vidange de la cuve en question, de l’eau arrive dans celle-ci pour procéder à la pousser du concentré restant et l’envoyer à la cuve sélectionnée de la zone 50.

I.5.4 Zone 50 (Reconstitution)



FigureI.24 : Vue de l’unité Reconstitution.

Cette zone regroupe :

- Trois Tanks (TR51, TR52, TR53).
- Trois agitateurs (50AG10, 50AG20, 50AG30) sur chaque tank.

- Deux capteurs de niveaux bas (50LSL10, 50LSL20, 50LSL30) et deux niveaux hauts (50LSH10, 50LSH20, 50LSH30).
- Soixante vanne (50V100, 50V101, 50V102, 50V103, 50V104, 50V105, 50V106, 50V107, 50V108, 50V111, 50V112, 50V113, 50V114, 50V200, 50V201, 50V202, 50V203, 50V204, 50V205, 50V206, 50V207, 50V208, 50V211, 50V212, 50V213, 50V214, 50V300, 50V301, 50V302, 50V303, 50V304, 50V305, 50V306, 50V307, 50V308, 50V310, 50V311, 50V312, 50V313, 50V314, 50V500, 50V501, 50V502, 50V503, 50V504, 50V505, 50V506, 50V600, 50V601, 50V602, 50V603, 50V604, 50V605, 50V606, 50V607, 50V608, 50V609, 50V610, 50V611, 50V620, 50V700, 50V701, 50V702, 50V703, 50V704, 50V705, 50V706).
- D'une sonde de température (50TT01).
- Trois débitmètres (50MT01, 50MT02, 50FT01).
- Cinq pompes, deux pompes de recirculation (50P20, 50P23) et cinq pompes d'envoi (50P10, 50P30, 50P22).
- Echangeur de chaleur (refroidissement avec l'eau glycolée).

Description du fonctionnement

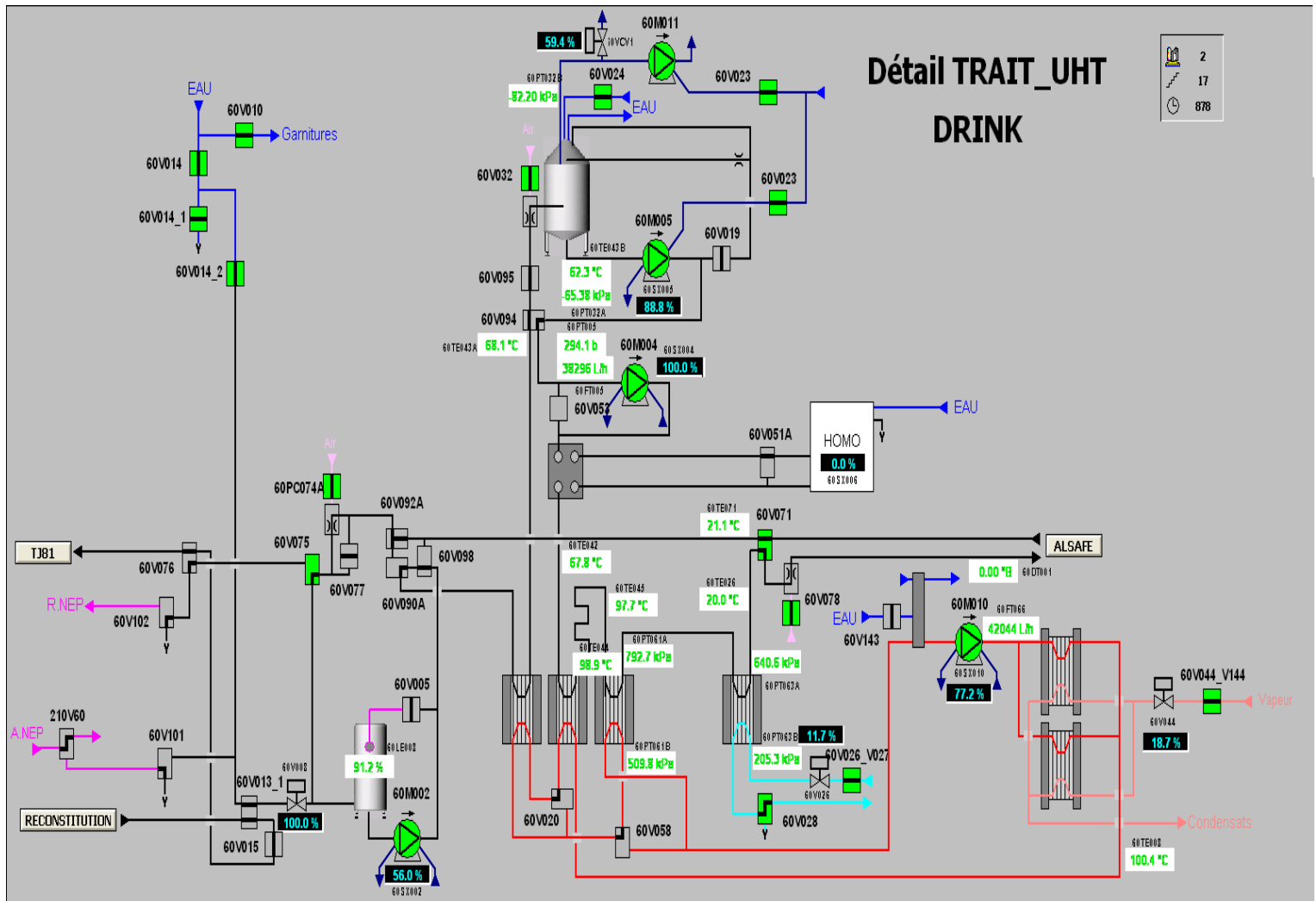
La zone 50 regroupe plusieurs connections avec le tank rejet, les tanks sirop, de l'ALMIX, du Tank concentré 41 et 41, et station NEP, et trois tanks de reconstitution (TR51, 52 et 53) qui sont muni d'agitateurs, d'une arrivé NEP et d'eau, de vannes de recirculation, de module de poudrage et d'une pompe d'envoi vers la zone 60 (DRINK).

Au lancement de la préparation, l'eau arrive dans l'un des tanks sélectionné à une température prédéfinie (refroidi avec l'eau glycolée, à l'aide d'un échangeur de chaleur), le rejet est incorporé de la zone 80, puis le dosage du sucre simple des cuves TS21 et/ou TS 22, le mixte est dosé de l'ALMIX, le concentré de fruits est transféré du TC41 ou TC42.

Une recirculation s'effectue à la fin de la préparation pour le contrôle du Brix et l'ajustement de la préparation, suivant les spécifications prédéfinies (selon la norme de l'entreprise), au finale la validation de la préparation s'effectue en mentionnant les paramètres physico-chimiques et finalisation de la recette.

Toutes ces phases s'effectuent automatiquement suivant la recette de formulations sélectionnée.

I.5.5 Zone 60 Pasteurisateur (DRINK)



FigureI.25 : Vue de l'unité Pasteurisateur.

Cette zone regroupe :

- Un bac de lancement
- Six échangeurs tubulaire (deux de préchauffage, deux autre de chauffage, un de pré-refroidissement et un refroidissement).
- D'un dégazeur.
- Cinq pompes, une de recirculation (60M004), trois pompes d'envoi (60M002, 60M005, 60M010) et une pompe à vide (60M011).
- Quatre vannes modulantes (60V008, 60V026, 60V044, ioVCV1).
- Trente-cinq vanne (60V005,60V010, 60V011, 60V013, 60V015, 60V014, 60V014_1, 60V014_2, 60V019, 60V020, 60V023, 60V024, 60V026_V027, 60V028, 60V032, 60V044_V144, 60V051, 60V052, 60V058, 60V071, 60V074,

60V075, 60V076, 60V077, 60V078, 60V090, 60V092, 60V094, 60V095, 60V098, 60V101, 60V102, 60V0143, 210V60).

- Huit sondes de température (60TE026, 60TE042, 60TE043B, 60TE043A, 60TE044, 60TE046, 60TE071, 60TE008).
- Sept manomètres (60PT032A, 60PT032B, 60PT005, 60PT061A, 60PT061B, 60PT063A, 60PT063B).

Description du fonctionnement

Le produit arrive au bac de lancement, puis une pompe expédie ce produit en passant par le préchauffage pour augmenter la température du produit à une valeur prédéfinie vers le dégazeur, à fin d'éliminer l'oxygène contenu dans le produit grâce à une agitation thermique des molécules sachant qu'une pompe à vide expulse ce gaz à l'extérieur. À la sortie de ce dégazeur il va continuer par le chauffage, ensuite il passe au pré-refroidissement à fin d'éviter un choc thermique. Juste après le produit passe au refroidissement et enfin il rejoint le tank aseptique.

Pour permettre les échanges de chaleurs au point de préchauffage, chauffage et pré-refroidissement de la vapeur passe dans un échangeur tubulaire pour chauffer l'eau qui passe dans l'échangeur tubulaire de chauffage, puis dans celui du préchauffage sachant qu'elle perd de la chaleur avant d'arriver à celui-ci, après elle passe dans celui du pré-refroidissement ou elle perd aussi de la chaleur et le cycle se répète à fin d'assurer les actions d'échange de chaleur (préchauffage, chauffage et pré-refroidissement)

Le refroidissement se fait à l'aide d'un échangeur tubulaire où passe le produit et de l'eau glycolé qui permet le refroidissement de celui-ci.

I.5.6 La Zone 70(ALSAFE)

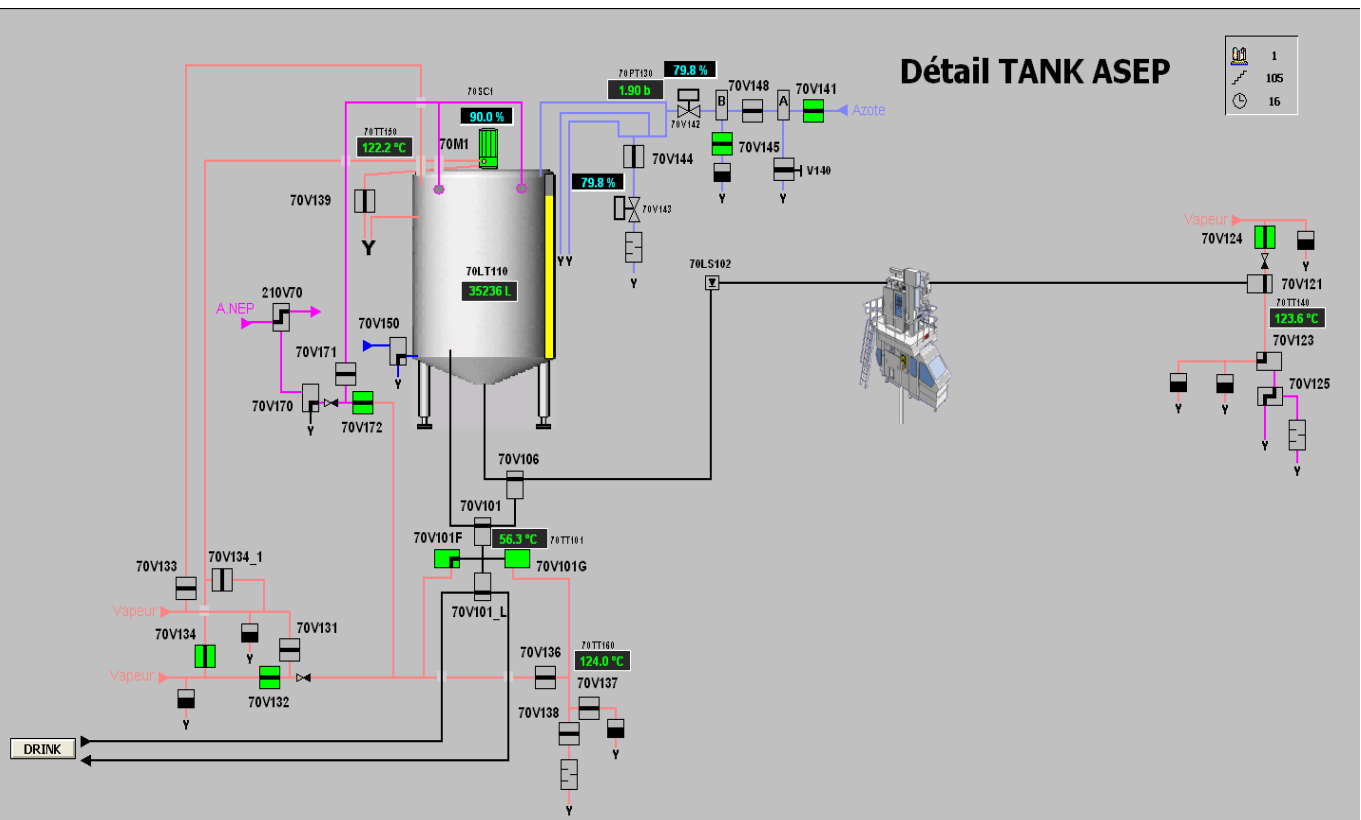


Figure I.26 : Vue de l'unité ALSAFE.

Cette zone regroupe :

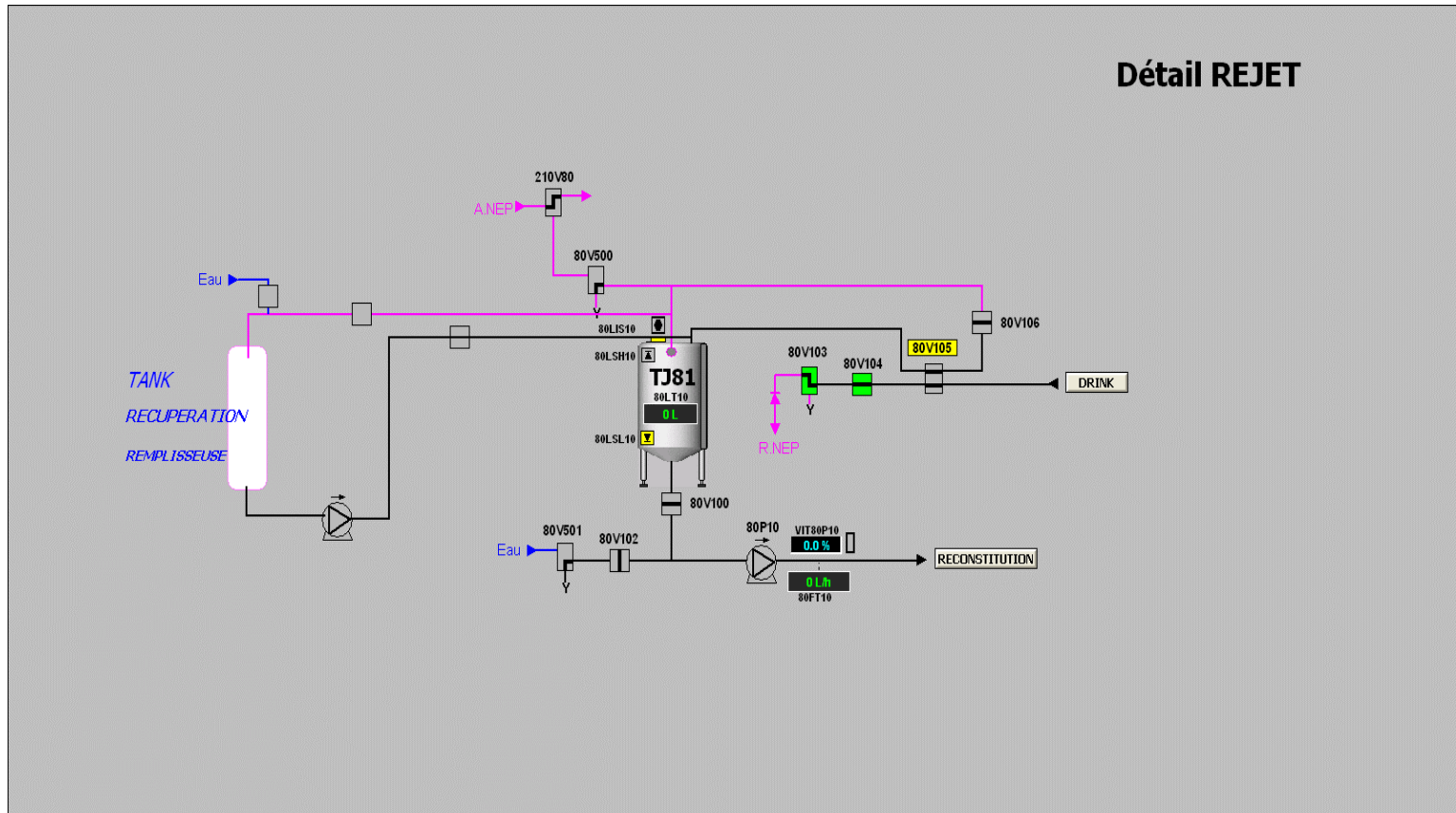
- Un tank aseptique.
- Vingt-sept vannes (70V101, 70V101F, 70V101_L, 70V101G, 70V106, 70V121, 70V123, 70V124, 70V125, 70V131, 70V132, 70V133, 70V134, 70V134_1, 70V136, 70V137, 70V138, 70V139, 70V141, 70V144, 70V145, 70V148, 70V150, 70V170, 70V171, 70V172, 210V70) ,Deux modulantes (70V142, 70V143) et une manuelle (V140).
- Quatre sondes de température (60TT101, 60TT140, 60TT150, 60TT160).
- Un manomètre (70PT130).
- Un Agitateur (70M1).

Description du fonctionnement

Le produit pasteurisé est envoyé au tank aseptique qui doit être stérile et sous pression d'azote qui est en fonction du niveau du produit. Ce tank est équipé d'une sonde de niveaux pour avoir le niveau maximum du produit a utilisé et un agitateur qui tourne à une vitesse qui dépend du niveau du produit. Ce système aussi est composé des vannes régulatrices pour la pression dans le tank, si la pression est au-dessous de la consigne la

vanne s'ouvre pour augmenter la pression si la pression est au-dessus de la consigne la vanne se ferme pour baisser la pression, on a aussi les barrières vapeur sécurisants le tank.

I.5.7 La Zone 80(rejet)



FigureI.27 : Vue de l'unité Rejet.

Cette zone regroupe :

- deux tanks, un rejet (TJ81) et un autre de récupération.
- Deux pompes d'envoi (80P10).
- Un Débitmètre (80FT10).
- Neuf vannes (80V100, 80V102, 80V103, 80V104, 80V105, 80V106, 80V500, 80V501, 210V80).
- Un capteur de niveaux bas (20LSL10) et un niveau haut (20LSH10).

Description du fonctionnement

Lorsqu'on pasteurise l'un des tanks de la zone50 (reconstitution) le produit va être transféré vers la zone60 (Drink) par la pompe de transfert drink pour le stocké dans le bac de lancement qui est auparavant plein avec 3000L d'eau, donc quand le produit arrive dans ce

- six capteurs de niveau (200LSH1, 200LSH2, 200LSH3, 200LSH.M1, 200LSL1, 200LSL4) sur chaque bac et tank.
- Trois sondes de température (200TE1, 200TE2, 200TE3).
- Trois pompes d'envoie (200M1, 200M2, 200M3).
- D'un échangeur à plaque.
- D'un débitmètre (200FT1).
- D'un conductimètre (200QE1).

Description du fonctionnement

Station de nettoyage en place ou **CIP (cleaning in incide place)**, s'occupe du nettoyage des circuits internes ; ou passe le produit en passant une solution pour enlever les résidus laissés après le passage du produit.

Le nettoyage s'effectue suivant trois étapes principales :

- Rinçage avec de l'eau : enlever résidus éliminables avec l'eau.
- Traitement avec les solutions nettoyantes (soit avec de la soude ou de l'acide) : pour éliminer les matières organique et minérale restantes.
- Rinçage final : pour éliminer les traces du produit chimique.

Le nettoyage commence en sélectionnant une cuve ou un tank, c'est grâce à une pompe qui envoie l'eau du bac de lancement de la station CIP vers la cuve ou le module sélectionné, après l'eau est évacuer vers l'égout.

Après le rinçage primaire, commence la phase de traitement avec les solutions nettoyantes déjà préparés à une concentration prédéfinie, ces dernier sont envoyés du tank de l'acide ou de la soude vers la cuve ou le module sélectionné, après fin de l'opération ; ces dernières sont récupérés jusqu'à une limite inférieur de concentration.

Au final la phase de rinçage qui suit le même chemin que le rinçage avec de l'eau.

I.5.9 Zone10 (de fabrication de sirop)

Cette zone n'est pas en service actuellement.

Contexte et Problématique

Le service production de la siroperie à manifester des difficultés dans la constations des défauts produit, dû à l'absence d'application de gestion d'historique des paramètres de production, ainsi une lenteur dans la recherche des résultats dans les registre saisis à la main. Tous ces facteurs engendrent une perte de temps, de ressources et un risque important dans le contrôle de qualité.

En effet, il est indispensable d'intégrer l'informatique pour l'historisation de ces paramètres de production. De ce fait, il serait utile de joindre les avantages qu'ont toutes ces techniques dans la réalisation d'un système de gestion d'historique pour le service siroperie. Pour cela nous devons répondre aux questions suivantes :

- Quelle sont les outils nécessaires pour mener à bien ce projet ?
- comment coordonner et sauvegarder ces paramètres ?

Conclusion

Dans ce chapitre nous avons présenté le service de production (siroperie), ainsi que les zones qui la constitue.

Ensuite nous avons procédé à la présentation de ces équipements et instruments, ainsi que leurs caractéristiques. Puis nous avons élaboré une analyse fonctionnelle du système de la siroperie qui nous facilitera la tâche dans l'identification des paramètres à sauvegarder.

Chapitre II

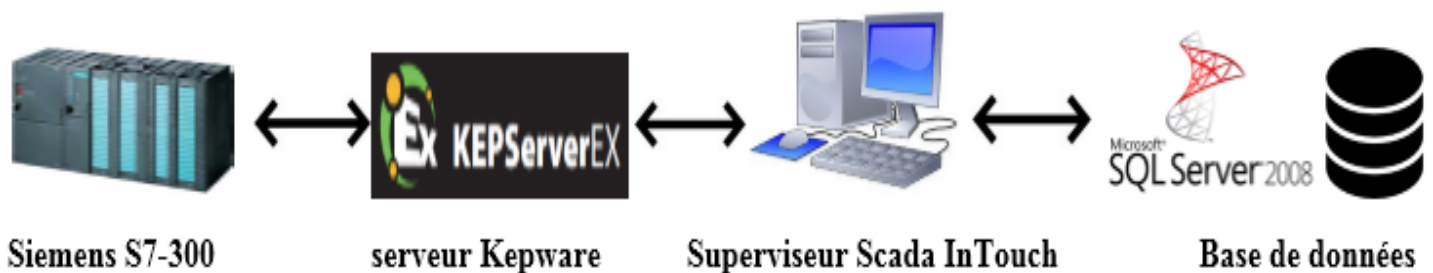
Généralité sur les Automates et logiciel associés et les différentes liaisons

Introduction

De nos jours la traçabilité des produits industriels joue un rôle très important dans le monde industriel, et plus particulièrement au sein des entreprises à caractère économique, pour améliorer l'ensemble des process de production, et garantir la qualité des produits et des services.

Automatiser un procédé et le supervisé c'est assurer l'acquisition de données et des paramètres de commande des processus par un dispositif technologique qui est l'automate programmable.

Dans ce qui suit nous allons présenter et voir quelque définition sur les automates programmables industriels d'une manière générale, et particulièrement l'automate S7-300, sur le système de communication KEPServer, puis sur logiciel de supervision InTouch et les bases de données SQL server, qui sont nécessaire à la réalisation de notre projet. Ensuite nous allons voir les différentes liaisons entre cette automate S7-300 et les logiciels (KEPServerEX, InTouch et Microsoft SQL Server).



FigureII.1 : Schéma explicatif.

II.1 Généralité sur l'Automate Programmable Industriel

II.1.1 Définition

Un automate programmable industriel (API) est un dispositif électronique possédant l'architecture d'un ordinateur (très proche de l'ordinateur) adapté aux milieux industriels destiné à la commande de processus par traitement séquentiel. Il envoie des ordres vers les actionneurs (partie opérative) à travers les pré-actionneurs (partie commande), à partir de données d'entrée (capteurs), des consignes et d'un programme informatique [1].

En résumé, l'automate constitue donc le cœur de l'automatisme. Il permet de contrôler, coordonner et d'agir sur la partie opérative c'est à dire la partie physique comme par exemple un robot, un bras manipulateur ou un tapis roulant.

II.1.2 Structure interne de l'Automate

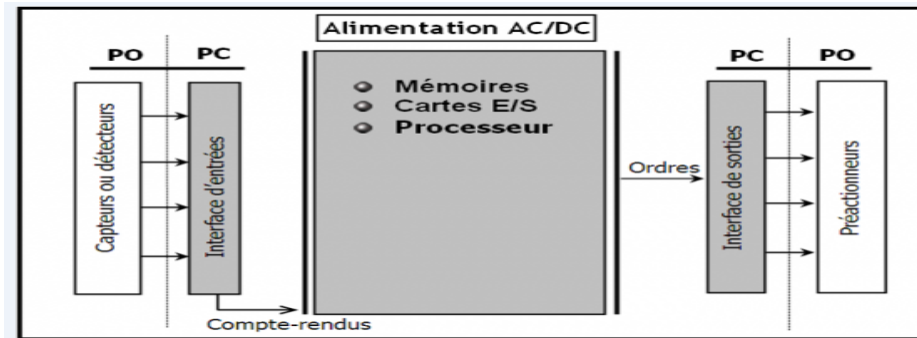
Les automates programmables comportent quatre parties principales :

- des mémoires.
- Un processeur plus des cartes d'E/S.

- Des interfaces d'E/S (IES).
- Une alimentation (220VAC, 24VCC).

Ces quatre parties sont reliées entre elles par des « bus » (ensemble de fils autorisant le passage des informations entre ces quatre secteurs de l'automate). Ces quatre parties réunies forment un ensemble compact appelé « automate ».

Un API peut se structurer de la manière suivante :



FigureII.2 : Structure d'un API.

II.1.3 Critères de choix d'un automate

Le choix d'un automate s'effectue suivant certains critères, comme :

- Capacité de traitement du processeur (vitesse, données, opération, temps réel...).
- Le nombre et la nature d'E/S.
- Les moyens de dialogue et le langage de programmation.
- La fiabilité, robustesse, immunité aux parasites.
- Qualité du service après-vente, durée de la garantie [3].

II.1.4 Les avantages de l'API

- Puissance-rapidité.
- Autonomie-Facilité de maintenance.
- Simplification du câblage.
- Modification du programme facile par rapport à la logique câblée.
- Fiabilité et énorme possibilité d'exploitation.

II.1.5 Les inconvénients

- Plantage : très rare heureusement, tout le système est bloqué ce qui peut engendrer quelques incidents.
- Cherté de l'appareil.

II.2 AUTOMATE PROGRAMMABLE SIMATIC S7-300

II.2.1 Définition

C'est un logiciel de commande modulaire pour des applications haut de gamme. Conçu pour des solutions dédiées au système manufacturier et constitue à ce titre une plate-forme

d'automatisation universelle pour les applications avec des architectures centralisées et décentralisées. Il est caractérisé par sa puissance, sa rapidité.

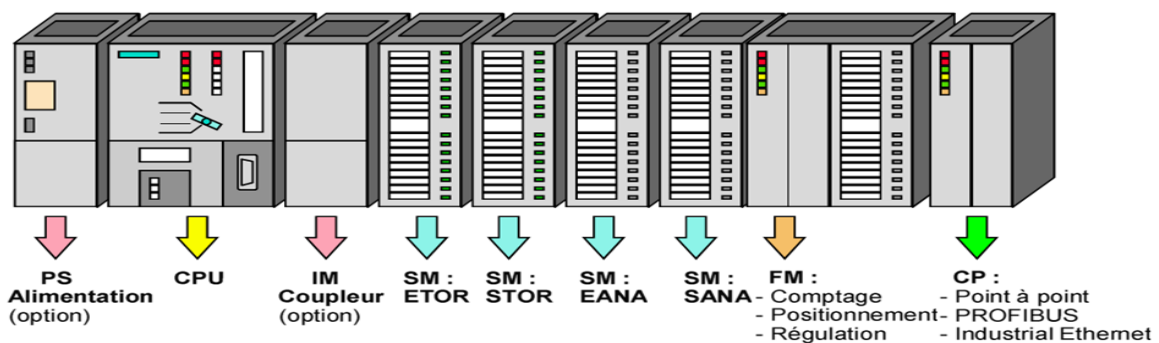


FigureII.3 : Vue externe de l'Automate S7-300.

II.2.2 Composants de S7-300

Le SIMATIC S7 -300 est un système d'automatisation modulaire offrant la gamme de modules suivants :

- Unités centrales CPU (central processing unit) : de capacités différentes, certaines avec d'entrées/sorties intégrées (ex : CPU314C) ou avec interface PROFIBUS intégrée (ex : CPU315-2DP)
- Modules d'alimentation PS (Power Supply) : Il transforme la tension alternative du réseau en tension de service DC 24V, et assure l'alimentation du S7-300 ainsi que l'alimentation externe pour les circuits de charges DC 24V.
- Modules d'extension IM (interface module) : pour configuration de plusieurs lignes du SIMATIC S7 -300
- Modules de signaux SM (Signal Module) : pour entrées et sorties numériques et analogiques
- Modules de fonction FM (Function Module) : pour fonctions spéciales (ex : pilotage d'un moteur pas à pas).
- Processeurs de communication CP: pour la connexion au réseau [16].



FigureII.4 : Vue général de l'Automate S7-300.

II.2.3 Les Interfaces

II.2.3.1 Interface MPI

Chaque CPU est équipée d'une interface MPI pour la connexion de la ligne de programmation (ex : PC Adapter). Celle-ci se trouve sous le volet de protection à l'avant de la CPU.

II.2.3.2 Interface Ethernet

L'interface Ethernet garantit des temps de transmission très courts, même pour des longs programmes. Interface Ethernet pour saisie externe des données pour la sauvegarde de toutes les données de tests importantes, nécessitées pour une évaluation plus approfondie et le protocole (option).

Pour les automates qui ne disposent pas de ports Ethernet en natif, un processeur de communication (CP) muni de port Ethernet fera l'affaire. Pour la gamme des automates S7-300, le processeur de communication CP343 pourra servir de coupleur Ethernet [15].

II.2.3.3 Interface PROFIBUS

PROFIBUS est un système de bus qui peut être mis en place aussi bien dans la couche liaison que pour des unités réseaux à faible nombre d'utilisateurs.

Pour le PROFIBUS, il y a 3 profils de protocoles qui peuvent être mis en place tous ensembles sur le même câble (RS 485 ou fibre optique) :

- **PROFIBUS FMS** (Fieldbus Message Spécification) : est adapté pour la communication d'automates entre eux dans de petites unités de réseau et pour la communication avec des appareils à champ par des interfaces FMS.
- **PROFIBUS DP** (Périphérie Décentralisée) : est le profil de protocole pour la connexion de périphériques décentralisés dans les domaines à champ, comme par ex. les modules ET 200 à temps de réaction très rapides.
- **PROFIBUS PA** (Process Automation) : est une extension compatible communication du PROFIBUS DP pour une technique de transmission, autorisant l'utilisation dans le domaine Ex. La technique de transmission du PROFIBUS PA respecte la norme standard internationale IEC 1158-2.

Dans le cadre de cette annexe, nous nous limiterons toutefois au PROFIBUS DP, que nous étudierons plus en détails, puisque c'est, avec le SIMATIC S7-300, le cas le plus fréquent d'utilisation et qu'il existe une interface intégrée spécialement dédiée [4].

II.3 Généralité sur KEPServerEx

II.3.1 Définition de KEPServer

KEPServerEX est un serveur OPC et une plate-forme simple d'utilisation contenant plus d'une centaine de drivers compatibles avec une large gamme d'automates, de bases de données et de systèmes. Il représente la nouvelle génération des technologies de communication Kepware.

KEPServer est devenu le standard dans la connectivité industrielle pour l'automatisme dans le monde [5].

II.3.2 Application client / serveur

Il est très courant d'entendre le terme "application client / serveur" utilisé dans de nombreuses disciplines et segments d'activité logiciels. Sur le marché industriel, il s'agit généralement du partage de données de fabrication ou de production entre diverses applications allant des logiciels d'interface machine humaine et des historiens de données, aux applications MES (Manufacturing Execution System) et ERP (Enterprise Resource Planning) de grande taille.

Indépendamment du secteur d'activité desservi, les applications client / serveur ont une chose en commun: une méthode standardisée de partage de données [12].

II.3.2.1 Architecture d'une Application clients/serveurs

Le modèle client-serveur. Une application est bâtie selon une architecture client-serveur lorsqu'elle est composée de deux programmes, coopérant l'un avec l'autre à la réalisation d'un même traitement. La première partie, appelée module client, est installée sur le poste de travail alors que la seconde, appelée module serveur, est implantée sur l'ordinateur (ou même des ordinateurs éventuellement situés dans des lieux géographiques différents) chargé de rendre le service (micro, mini ou grand système).

II.3.3 OPC

OLE for Process Control (liaison d'objets et encastrement pour le contrôle de processus). Est une technique apparue en 1994 et destinée à l'interopérabilité des systèmes industriels. Il ne s'agit pas d'un protocole de communication mais d'une technique basée sur les techniques OLE, COM, et DCOM développées par Microsoft pour sa famille de systèmes d'exploitation Windows.

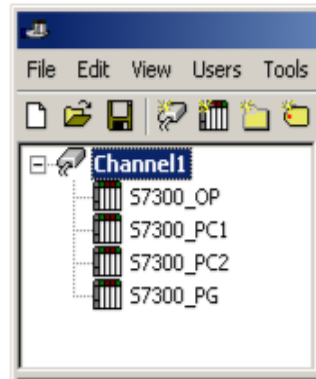
Un objectif majeur de la Fondation OPC et la spécification d'accès aux données était d'éliminer la nécessité pour les fournisseurs d'applications clientes d'élaborer leur propre ensemble propriétaire de pilotes de communication à fin de développer des d'applications clientes et se concentrer sur l'amélioration continue de son produit de base sans l'effort perturbateur nécessaire pour traiter les problèmes et les besoins en matière de communication.

II.3.4 Optimisation des communications Ethernet TCP / IP Siemens

Le pilote Ethernet TCP / IP de Siemens fournit un moyen fiable de connecter des périphériques Ethernet TCP / IP Siemens à des applications client OPC, y compris HMI, SCADA, Historian, MES, ERP et d'innombrables applications personnalisées. Il est destiné à être utilisé avec les automates Siemens S7-200, 300, 400 et 1200.

KEPServerEX se réfère à des protocoles de communication comme Siemens TCP / IP Ethernet en tant que canal. Chaque canal défini dans l'application représente un chemin d'exécution distinct dans le serveur. Une fois qu'un canal a été défini, une série d'appareils peut ensuite être définie sous ce canal. Chacun de ces périphériques représente un seul contrôleur Ethernet TCP / IP Siemens à partir duquel les données seront collectées. Un exemple de

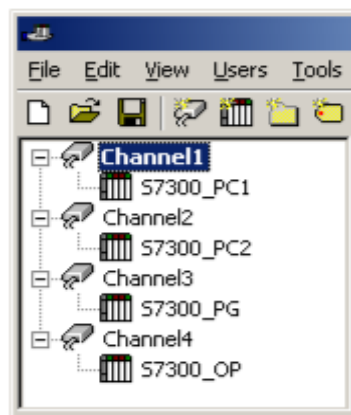
fonctionnement de l'application lors de la configuration à l'aide d'un seul canal est illustré ci-dessous.



FigureII.5 : Exemple d'une configuration à l'aide d'un seul canal.

Chaque périphérique apparaît sous un seul canal Ethernet TCP / IP Siemens. Dans cette configuration, le pilote doit passer d'un périphérique à l'autre le plus rapidement possible pour collecter des informations à un rythme effectif.

Le pilote Ethernet TCP / IP Siemens peut définir jusqu'à 256 canaux. L'utilisation de plusieurs canaux distribue la charge de travail de collecte de données en émettant simultanément plusieurs requêtes sur le réseau. Un exemple de fonctionnement de la même application lors de la configuration à l'aide de plusieurs canaux est présenté ci-dessous.



FigureII.6 : Exemple d'une configuration à l'aide de plusieurs canaux.

Chaque périphérique a maintenant été défini sous sa propre chaîne. Dans cette nouvelle configuration, un seul chemin d'exécution est dédié à la tâche de collecte des données de chaque périphérique [12].

II.4 Généralité sur le système SCADA « InTouch »

II.4.1 Le logiciel InTouch

InTouch, de Wonderware est une interface IHM graphique des données d'usine et de procédé. Il permet de mettre en œuvre et déployer des applications de dialogue homme-machine

sous Windows, totalement distribuées et intégrées avec les autres applications de la suite ArcestrA: automatisation, traçabilité, historisation des données, pilotage de Batch et visualisation via Internet. Il saisit, affiche et archive des données de procédé et les présente à l'opérateur, idéalement dans un format facile à utiliser. C'est une puissante interface opérateur qui prend des données des dispositifs de contrôles/commande (API, capteurs intelligents, etc...) et les affiche pour qu'elles soient utiles à un opérateur. On retrouve les applications d'InTouch dans tous les environnements industriels : industrie manufacturière, process ou tertiaire, pour le contrôle-commande, la supervision ou l'acquisition de données.

InTouch, tout en étant puissant, demeure simple à utiliser une fois qu'on comprend les concepts de base [14].

II.4.2 InTouch HMI : visualisation du processus de fabrication

Une application de type HMI (Human Machine Interface) est un développement logiciel offrant une représentation graphique d'un environnement de fabrication. Les outils, les matériels et les processus utilisés pour la création d'un produit sont présentés en tant qu'éléments visuels dans des fenêtres d'une application HMI. Les opérateurs d'usine se servent de cette interface graphique pour surveiller et gérer les processus de fabrication [7].

II.4.3 Création d'application

WindowMaker permet de créer une interface visuelle pour les applications HMI. WindowMaker dispose d'outils graphiques, d'un langage de scripts, d'un outil gestionnaire de variables qui définit le comportement des objets affichés par les fenêtres de l'application.

WindowMaker permet de créer des variables représentant des points de données associées à des objets dans les fenêtres. Une information provenant d'un processus de fabrication est finalement associée à une valeur de variable. L'application peut utiliser cette information de la variable pour surveiller les alarmes, pour créer des courbes ou pour déterminer le comportement de l'application pendant l'exploitation.

L'illustration suivante montre certains des outils de WindowMaker permettant de créer des applications HMI.

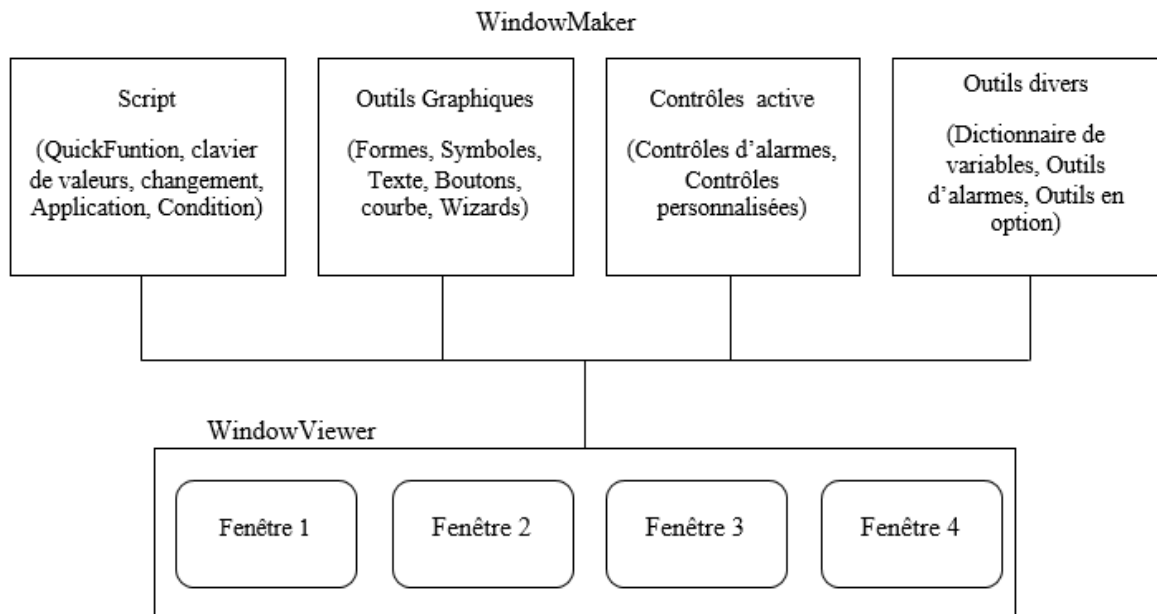


Figure II.7 : Les outils de WindowMaker permettant de créer des applications HMI.

II.4.4 Gestion des applications

Après avoir créé une application, on doit régulièrement gérer l’environnement InTouch. La gestion d'applications, depuis le Gestionnaire d'applications.

Comme son nom le suggère, le Gestionnaire d'applications permet de créer et de gérer des applications InTouch. L’environnement de développement d'applications, appelé WindowMaker, contient un ensemble d'outils graphiques et de développement pour créer vos applications. Vous exécutez vos applications à l'aide de WindowViewer comme on le montre dans la figure ci-dessous [7] :

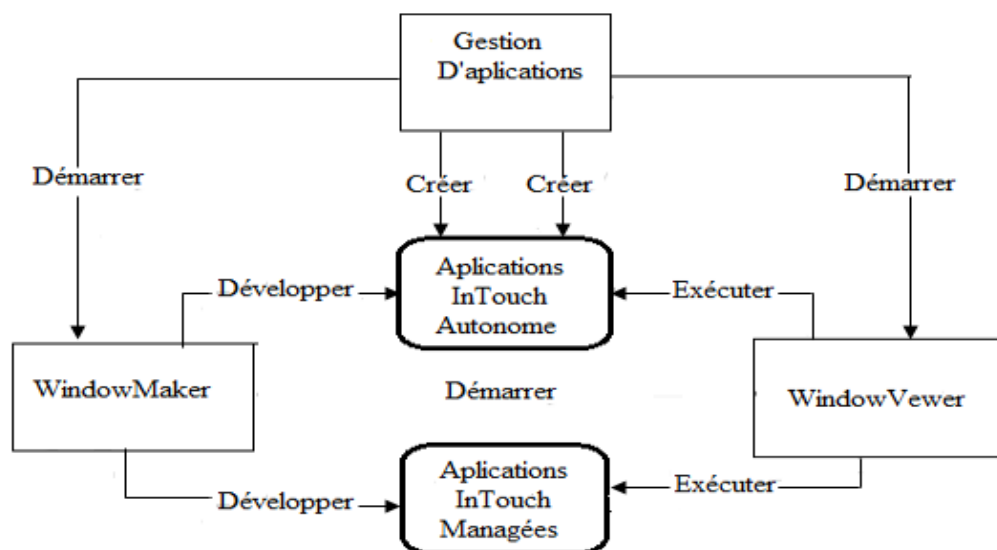


Figure II.8 : Démonstration comment les applications InTouch sont créées puis administrées avec le Gestionnaire d'applications.

II.4.5 Gestion de données

Après avoir créé une application InTouch, on définit les variables utilisées dans l'application à l'aide du dictionnaire de variables, qui est un utilitaire de WindowMaker.

II.4.6 Type de données

On peut attribuer aux variables des applications jusqu'à quatre types différents, selon la nature des données du processus associé. Les types spécifiques inclus par le Dictionnaire de variables pour les données sont de : type entier, type réel, type discret et type message. Le Dictionnaire de variables inclut des types de données pour les variables mémoire, les variables d'E/S et les variables indirectes.

II.4.7 Propriétés des variables

Le Dictionnaire de variables permet de créer des variables utilisées dans une application InTouch. L'illustration suivante montre la boîte de dialogue du Dictionnaire de variables avec toutes les options permettant de définir les propriétés d'une variable d'E/S.

Dictionnaire de variables

Principal Détails Alarmes Détails et alarmes Membres

Nouveau Restaurer Supprimer Enregistrer << Sélectionner... >> Annuler Fermer

Variable : 60TE043A Type... E/S réel

Groupe... TADRINK Lecture seule Lecture/écriture

Commentaire : Temperature 60TE043A

Enregis. données Enreg. événements Valeur mémorisée Paramètres mémorisés

Valeur initiale : 0 Val. UM min. : 0 Val. UM max. : 150

Bande morte : 0 Val. brute min. : 0 Val. brute max. : 150

Unités de mesure : °C Bande morte enreg. : 0 Conversion Linéaire Racine carrée

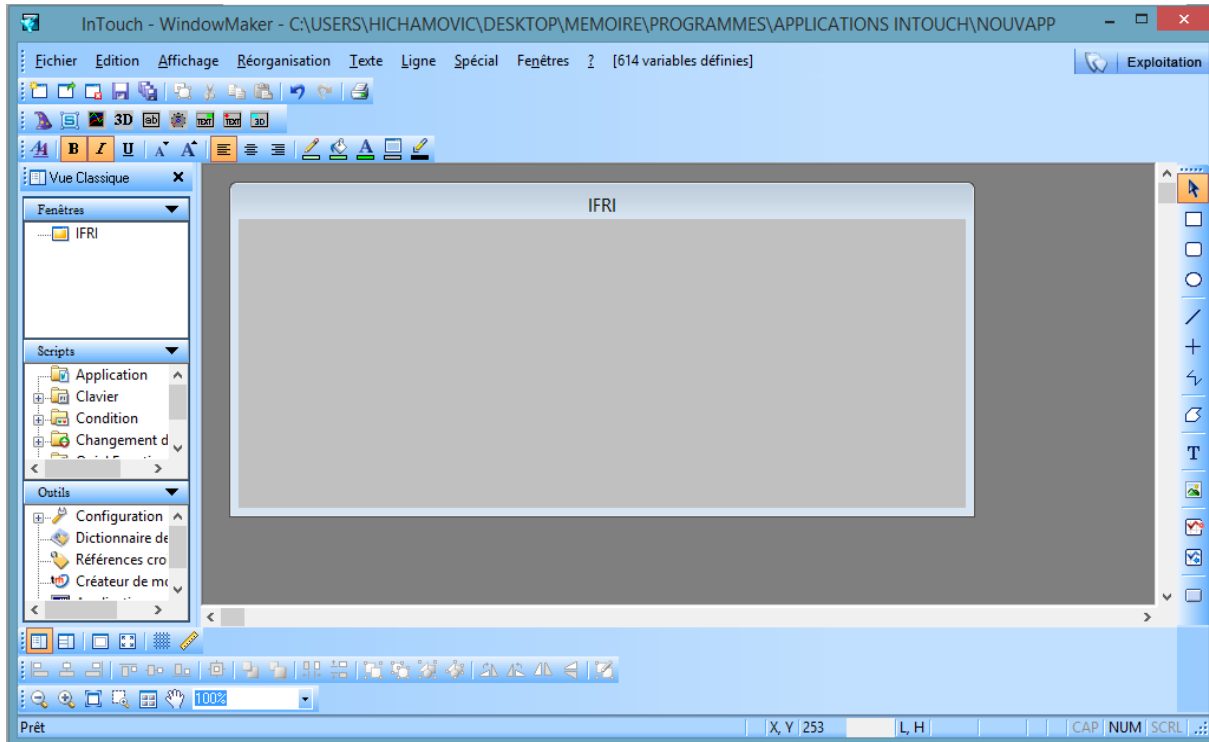
Nom d'accès... API_TADRINK

Elément : DB550.KG472 Utiliser le nom de variable comme nom d'élément

FigureII.9 : la boîte de dialogue du Dictionnaire de variables

II.4.8 Fenêtres InTouch

Une fenêtre d'application est le conteneur d'un ou plusieurs objets graphiques utilisés pour modéliser des processus de production. Les fenêtres InTouch sont des panneaux visuels contenant l'interface utilisateur de votre application InTouch.



FigureII.10 : les fenêtres de WindowMaker.

II.4.9 Points fort du logiciel InTouch

- Superbes graphiques industriels vectoriels prêts à l'emploi.
- Adapté à tout type d'application (monoposte, panel opérateur, application en réseau...) et toute taille d'application.
- Accroît l'efficacité des opérateurs avec une meilleure connaissance de la situation.
- Permet de prendre de meilleures décisions en temps réel.
- Augmente la productivité opérationnelle et d'ingénierie.
- Accélère la formation des opérateurs.
- Permet d'accéder en toute sécurité au système d'information à partir de n'importe quel équipement (PC, tablette, téléphone), n'importe quel endroit et à n'importe quelle heure [6].

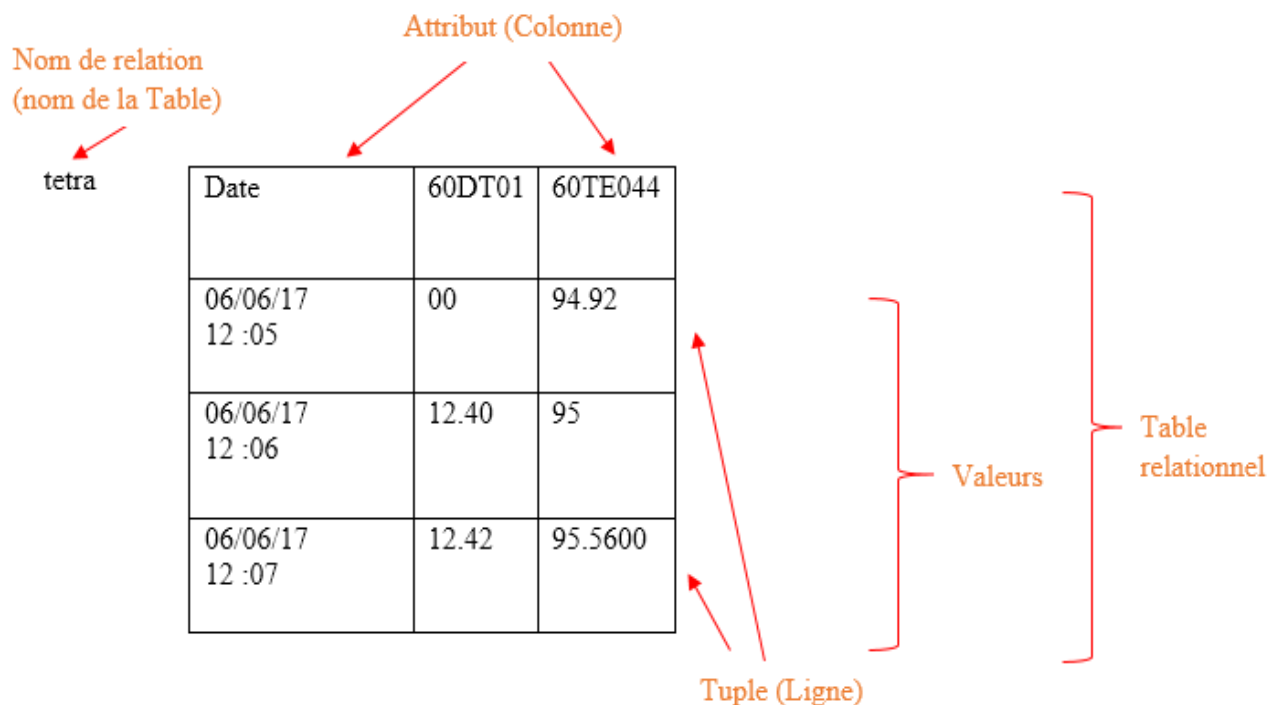
II.5 Généralité sur les bases de données

II.5.1 Définition d'une base de données

Une base de données, usuellement abrégée en BD ou BDD, est un ensemble de données structuré et organisé modélisant les objets d'une partie du monde réel, permettant le stockage de grandes quantités d'informations afin d'en faciliter l'exploitation (ajout, mise à jour, recherche de donnée..), et dans un temps record, mémorisées sur des supports accessible par un ordinateur, et servent de support à une application informatique. [10]

II.5.2 Les bases de données relationnelles

Une base de données relationnelle est une base de données où l'information est organisée dans des tableaux à deux dimensions appelés des relations ou tables, selon le modèle introduit par Edgar F. Codd en 1970. Selon ce modèle relationnel, une base de données consiste en une ou plusieurs relations. Les lignes de ces relations sont appelées des n-uplets ou Tuple ou enregistrements. Les colonnes sont appelées des attributs.



FigureII.11 : Modèle de données relationnel.

II.5.3. Les systèmes de gestion de bases de données (SGBD)

II.5.3.1 Notion d'un SGBD

Un système de gestion de base de données est un logiciel intermédiaire entre les utilisateurs et les bases de données. Destiné à stocker et à partager des informations dans une base de données, en garantissant la qualité, la pérennité et la confidentialité des informations, tout en cachant la complexité des opérations.

Un SGBD permet d'inscrire, de retrouver, de modifier, de trier, de transformer ou d'imprimer les informations de la base de données. Il permet d'effectuer des comptes rendus des

informations enregistrées et comporte des mécanismes pour assurer la cohérence des informations, éviter des pertes d'informations dues à des pannes, assurer la confidentialité et permettre son utilisation par d'autres logiciels. [12]

II.5.3.2 Objectifs des SGBD

Parmi les principaux objectifs que doivent remplir les SGBD nous citons :

- **Indépendance physique des données** : la représentation interne des données et les méthodes d'accès au système de fichier doivent être transparentes pour l'utilisateur.
- **Indépendance logique des données** : Un même ensemble de donnée peut être vu différemment par des utilisateurs différents. Toutes ces visions personnelles des donnée sont définit dans le schéma externe et doivent être intégrées dans une vision globale appelée schéma conceptuel. Une réorganisation, même profonde, du schéma conceptuel ne doit pas avoir d'impact sur le schéma externe.
- **Accès aux données** : L'accès aux données se fait par l'intermédiaire d'un Langage de Manipulation de Données (DML). IL est crucial que ce langage permette d'obtenir des réponses aux requêtes en un temps raisonnable. Le DML doit donc être optimisé, minimiser le nombre d'accès disque, et tout cela des façons totalement transparente pour l'utilisateur.
- **Administration centralisée des données** : Toutes les données doivent être centralisées dans un réservoir unique commun à toutes les applications. En effet, des visions différentes des données (entre autre) se résolvent plus facilement si les données sont administrées de façon centralisée.
- **Non-redondance des données** : Afin d'éviter les problèmes lors des mises à jour, chaque information ne doit être représentée qu'une seul fois dans la base.
- **Cohérence des données** : Les données son soumise à un certain nombre de contraintes d'intégrité qui définissent un état cohérent de la base. Elles doivent pouvoir être exprimées simplement, et vérifiées automatiquement à chaque insertion, modification ou suppression de données. Les contraintes d'intégrité sont décrites dans le Langage de Description de Données (LDD).
- **Partage des données** : Il s'agit de permettre à plusieurs utilisateurs d'accéder aux mêmes données au même moment de manière transparente. Si ce problème est simple à résoudre quand il s'agit uniquement d'interrogations, cela ne l'est plus quand il s'agit de modifications dans un contexte multi-utilisateur. En effet, il faut permettre à plusieurs utilisateurs de modifier la même donnée « en même temps » tout en assurant un résultat d'interrogation cohérent pour un utilisateur qui consulterait cette même donnée.
- **Sécurité des données** : Les données doivent pouvoir être protégées contre les accès non autorisés. Pour cela, il faut pouvoir associer à chaque utilisateur des droits d'accès aux données.
- **Résistance aux pannes** : Que se passe-t-il si une panne survient au milieu d'une modification, si certains fichiers contenant les données deviennent illisibles ? Il faut

pouvoir récupérer une base dans un état sain. Ainsi, après une panne intervenant au milieu d'une modification deux solutions sont possibles : soit récupérer les données dans l'état dans lequel elles étaient avant la modification, soit terminer l'opération interrompue. [12]

II.5.3.3 Les fonctions des SGBD

Parmi les fonctions essentielles d'un SGBD on a :

- Description des données.
- Recherche des données.
- Mise à jour des données.
- Transformation des données.
- Contrôle de l'intégrité des données.
- Gestion de transaction et sécurité [11].

II.5.4 langage SQL

Créé en 1974, normalisé depuis 1986, le langage est reconnu par la grande majorité des systèmes de gestion de bases de données relationnelles (SGBDR) du marché.

Le **SQL (langage de requête structurée)** est un langage informatique normalisé servant à exploiter des bases de données relationnelles.

Les instructions du langage SQL couvrent 4 domaines et sont regroupées en fonction de leur utilité et des entités manipulées.

- **Le langage de définition des données (DDL)** est le langage de manipulation de la structure de la base de données. Le DDL permet de créer, modifier et supprimer des objets comme des bases de données, des tables, des index, des contraintes...

Les instructions du DDL sont : **CREATE, ALTER, DROP, RENAME, TRUNCATE.**

- **Le langage de manipulation des données (DML)** rassemble l'ensemble des commandes de manipulation des données dans une base de données. Le DML permet l'extraction, l'ajout, la suppression et la modification des données de la base de données.

Les instructions du DML sont : **SELECT, INSERT, UPDATE, DELETE.**

- **Le langage de contrôle d'accès (DCL)** s'occupe de gérer les privilèges accordés aux utilisateurs, comme les droits d'accès aux tables.

Les instructions du DCL sont : **GRANT, REVOKE.**

- **Le langage de contrôle de transaction (TCL)** gère les modifications faites par le DDL et le DML, c'est-à-dire les caractéristiques des transactions et la validation et l'annulation des modifications.

Les instructions du TCL sont : **COMMIT, SAVEPOINT, ROLLBACK, SET TRANSACTION.** [12]

II.5.5 Le langage T-SQL

Microsoft Transact SQL est un langage de requêtes amélioré par rapport au SQL dans il reprend les bases.

Trois catégories d'instruction composent ce langage :

- **Le langage de définition des données (DDL)** permet la création, la modification et suppression des objets du SQL (TABLES, INDEX, VUES, PROCESURES,...).
- **Le langage de manipulation des données (DML)** fournissant les instructions de création, de mise à jour et de suppression et d'extraction des données stockées
- **Le langage de contrôle d'accès (DCL)** pour la gestion des accès aux données, des transactions et de la configuration des sessions et des bases.

De plus, le Transact SQL prend en compte les fonctionnalités procédurales telles que la gestion des variables, les structures de contrôles de flux, les curseurs, et les lots d'instruction. C'est donc un langage complet qui comporte des instructions, qui manipule des objets SQL, qui admet la programmabilité et qui utilise des expressions. [11]

II.5.6 Microsoft SQL Server 2008

Microsoft SQL Server est un SGBD développé et commercialisé par la société Microsoft , il fonctionne sous les OS Windows, Linux, Mac OS.

Microsoft SQL Server est un serveur de base de données relationnelle et à ce titre, il fournit tous les éléments pour stocker de façon structurée les données mais aussi les outils nécessaires pour travailler avec les données au travers le SQL. Avec le Transact SQL (T-SQL) il est également possible de définir des traitements procéduraux directement dans la base de données. Ces traitements vont pouvoir être utilisables par tous les utilisateurs de la base sous réserve qu'ils possèdent les privilèges nécessaires. Il est possible de conserver la définition de ces traitements et de les rendre paramétrables par l'intermédiaire de la création de procédures et de fonctions.

Des traitements procéduraux pourront également être mis en place pour définir des contraintes d'intégrité complexes, il s'agira alors de triggers ou déclencheurs de base de données. [11]

II.6 Les différentes liaisons

II.6. Liaison Siemens S7-300 avec KEPServer

La liaison se fait via un câble Ethernet qui relie l'automate au serveur centrale, puis KEPServer l'identifie à l'aide du pilote **Ethernet TCP / IP Siemens**.

La connexion sous KEPServer se fait comme suit :

On commence par créer un new Channel et la nomme comme le montre la figure ci-dessous :

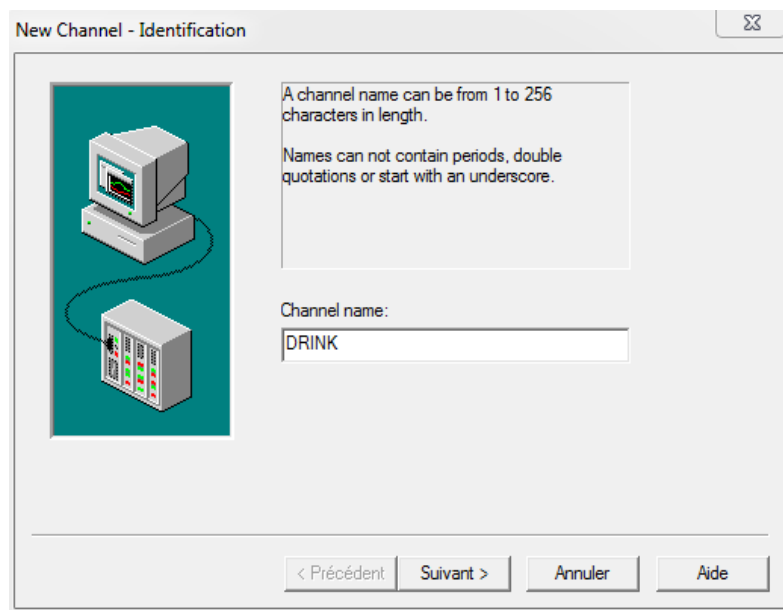


Figure II.12 : Capture de fenêtre de création d'un nouveau Channel.

Puis on clique sur suivant pour sélectionner le type de pilote de périphérique comme démontré ci-dessous :

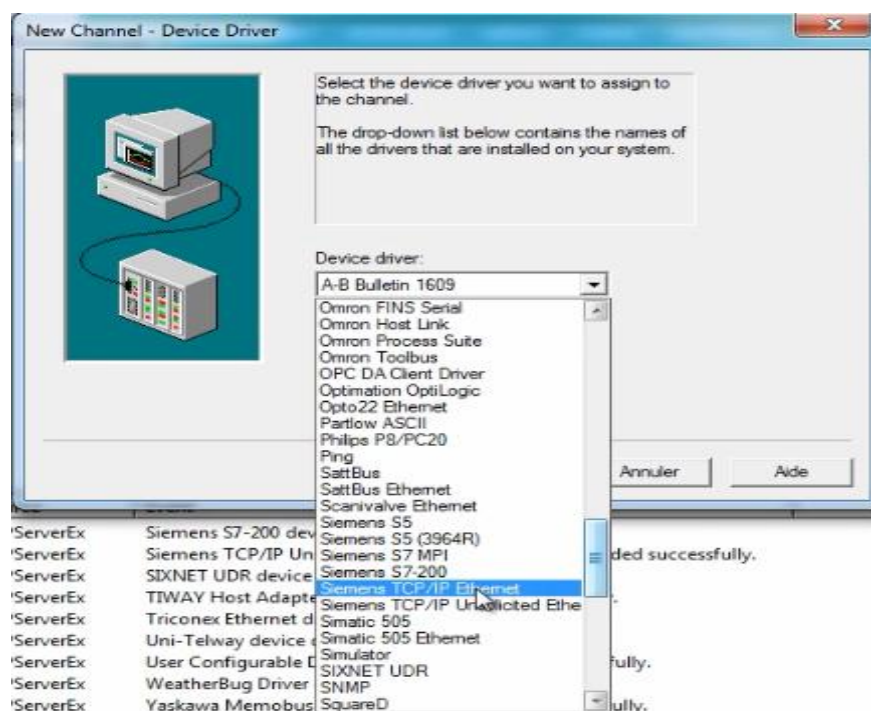
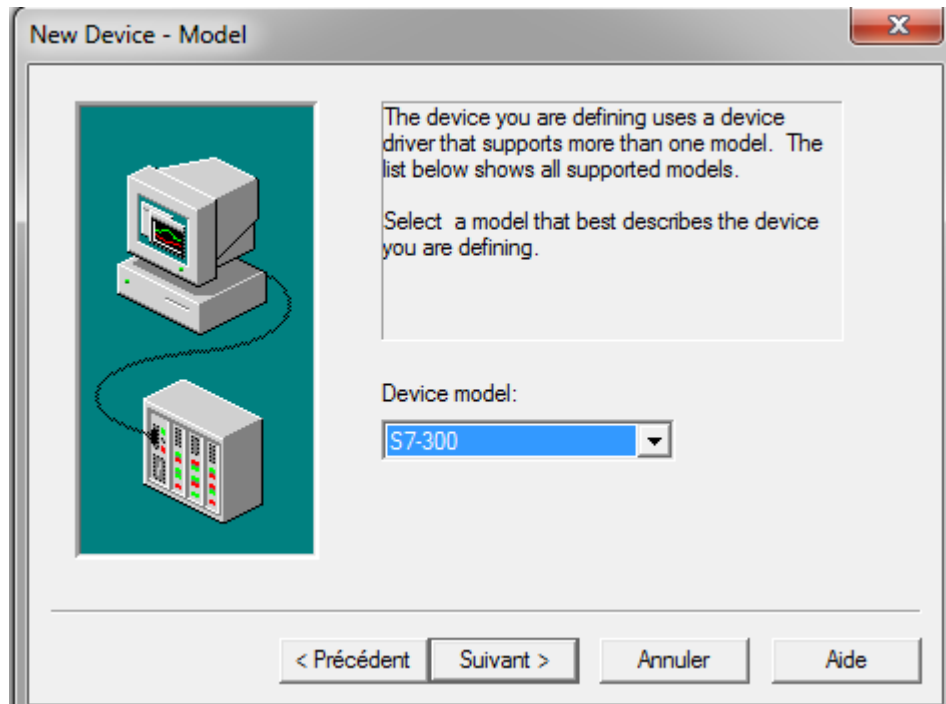


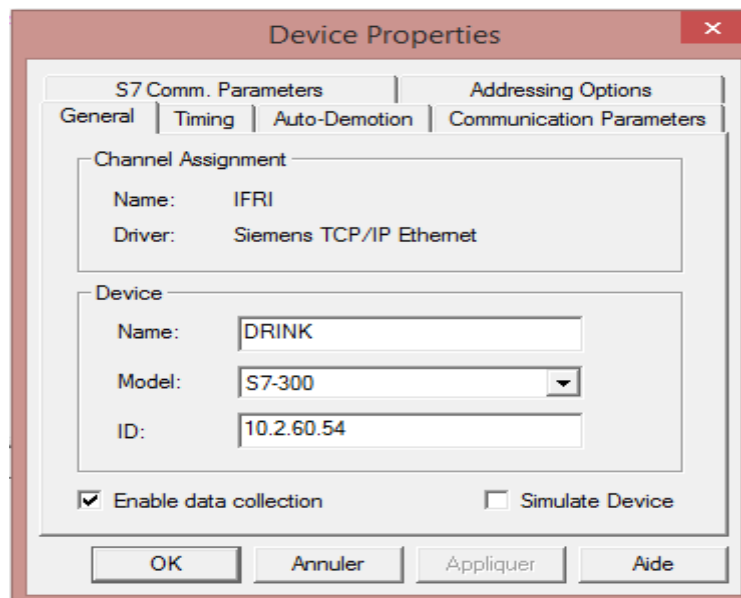
Figure II.13 : Capture de sélection de pilote de périphérique.

Et on continue en nommant ce périphérique, après on choisit le model du périphérique comme démontrée ci-dessous :



FigureII.14 : vue du choix du périphérique.

Puis on introduit ID du périphérique et on termine sur l’affichage de la fenêtre suivante qui décrit toutes les informations du Channel créer.



FigureII.15 : Capture qui décrit toute les informations du Channel créé.

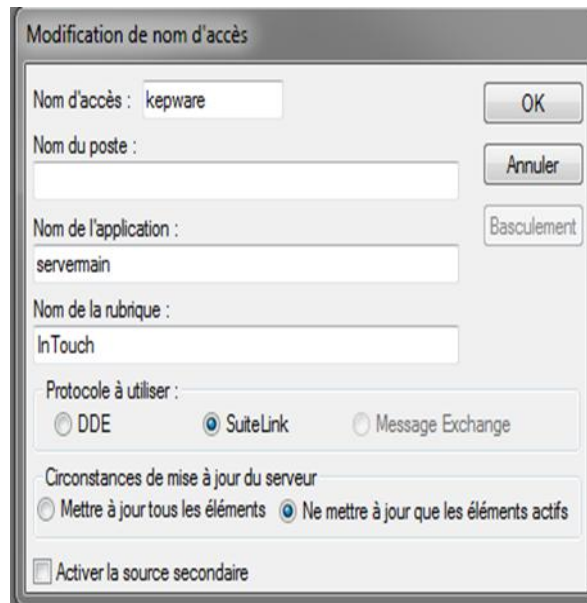
II. Liaison SCADA InTouch avec KEPServer

La liaison se fait via le server OPC. Cela se fait par plusieurs étapes :

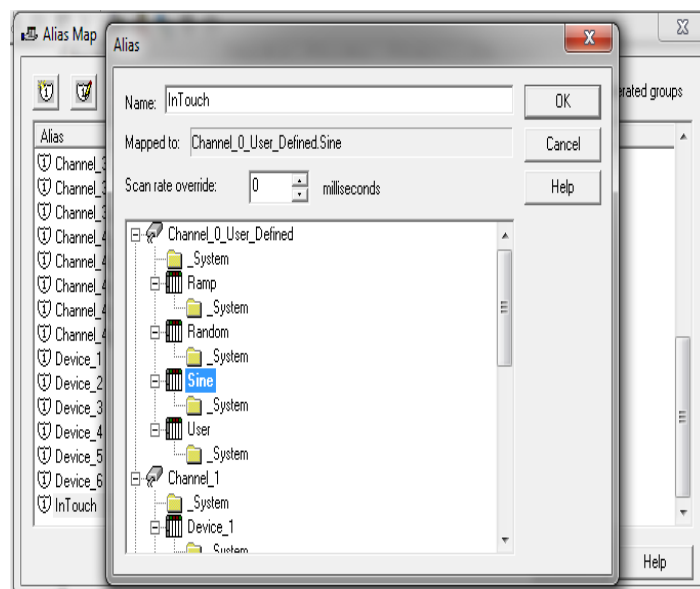
Premièrement on commence par cocher la case ‘enable FastDDE/SuiteLink connections to the server’ dans les options Tools de KEPServer, après on introduit le nom d’accès et le nom de l’application dans les paramètres spécial d’InTouch.

Deuxièmement nous utilisons la carte d’alias fournie de KEPServer ou nous pouvons créer une nouvelle facile à retenir « InTouch » et on sélectionne l’appareil.

Comme il est montré dans les figures ci-dessous :

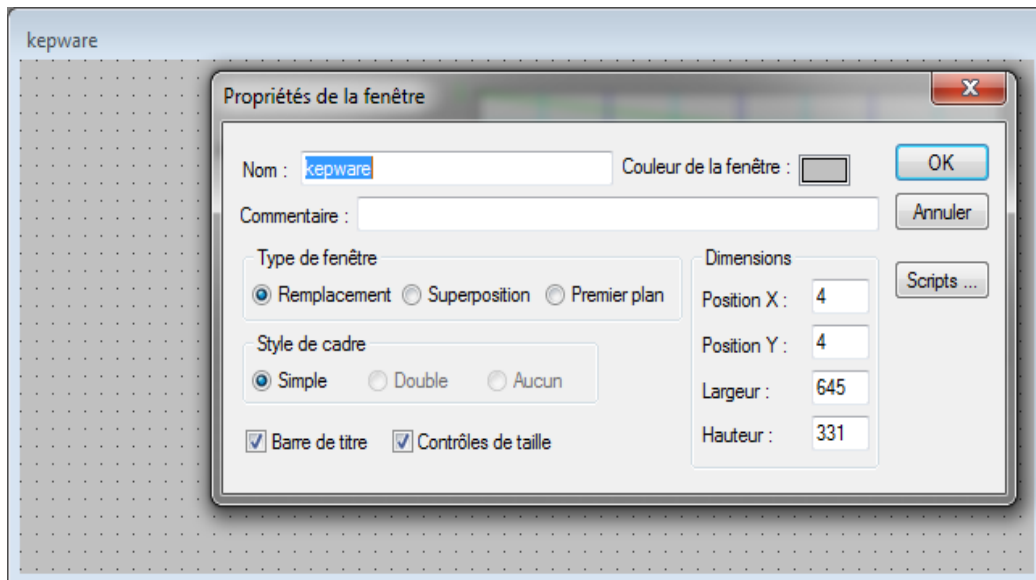


FigureII.16: Capture de modification de nom d’accès



FigureII.17: vue d’une fenêtre création d’une de carte d’alias

Finalement on va créer une nouvelle fenêtre dans InTouch sous le nom Kepware par exemple comme dans la figure ci-dessous :



FigureII.18 : Capture d'une fenêtre InTouch

III.3 Liaison InTouch avec SQL Server 2008

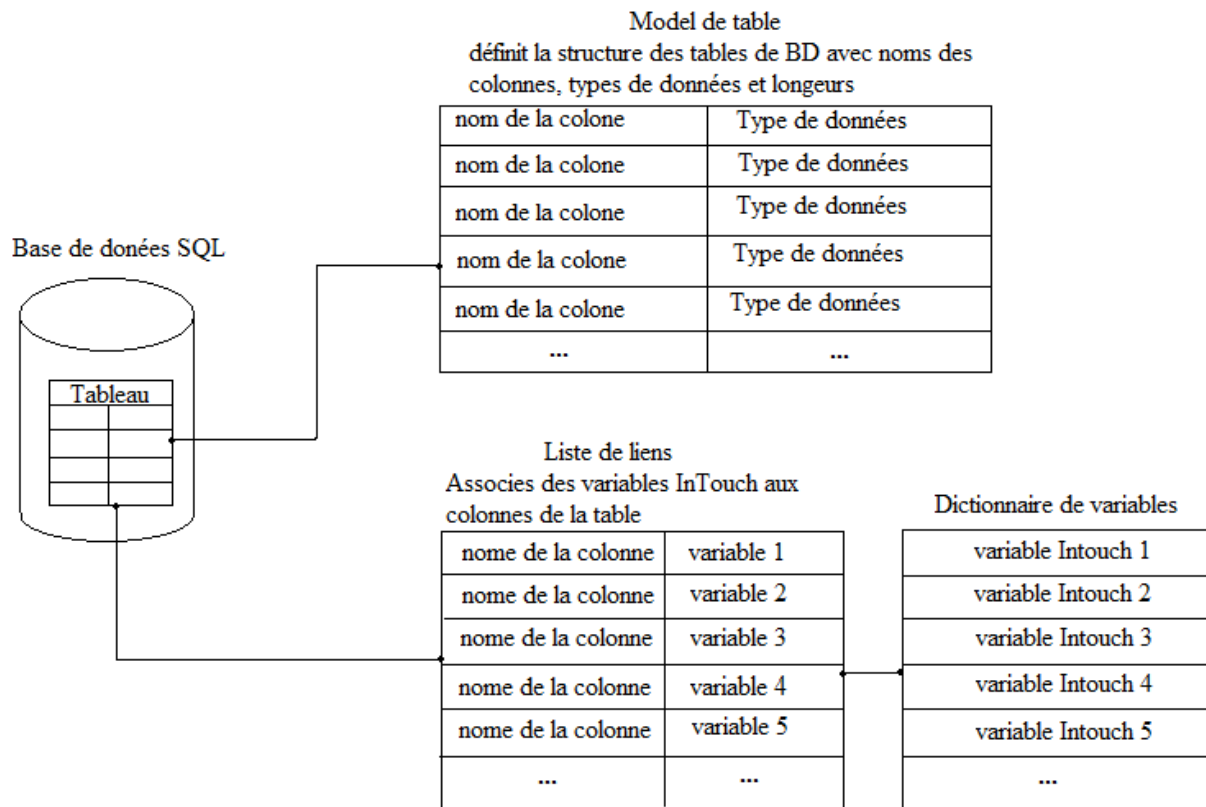
La liaison sera établie entre InTouch et SQL Server grâce à SQL Access Manager suite à des requêtes formulées dans des scripts sur InTouch qui permet d'accéder, de modifier, de créer et de supprimer des tables de base de données.

Le programme SQL Access Manager crée et associe les colonnes d'une base de données avec des variables InTouch. Ce processus d'association des colonnes d'une base de données avec des variables InTouch est appelé une «liaison » (binding, en anglais). La liaison des variables InTouch aux colonnes de la base de données permet à SQL Access Manager d'exploiter directement les données InTouch conservées dans la base de données.

II.7 Association de variables InTouch aux colonnes de la base de données

Des variables InTouch peuvent être associées aux colonnes de la base de données. Ceci s'effectue avec une liste de liens. La plupart des fonctions SQL Access utilisent la liste de liens pour donner à des variables InTouch l'accès aux données des tables de la base de données.

Une liste de liens associe les colonnes de la base de données aux variables contenues dans le dictionnaire de variables d'InTouch. Une liste de liens contient également un modèle de table, décrivant le format des tables de la base de données.



FigureII.19 : Capture représentant une liste de liens associe les colonnes de la base de données aux variables InTouch.

Quand un script contenant les fonctions `SQLInsert()`, `SQLSelect()` ou `SQLUpdate()` est exécuter, la liste des liens est mise à jour afin de spécifier les variables InTouch qui sont utilisées, et les colonnes dans la table, qui sont associées à ces variables.

Les actions SQL sont synchrones. Quand une fonction de base de données est exécutée dans un QuickScript, à partir d'une application InTouch, le contrôle ne revient à InTouch qu'une fois que l'action demandée par la fonction a été complétée sur la base de données.

Les fonctions SQL Access respectent des normes de ponctuation indiquant le type des arguments associé à une fonction. Lorsqu'une chaîne est transmise comme argument, encadrée par des guillemets ("Arg1"), le script utilise la chaîne littérale exacte. Si des guillemets ne sont pas utilisés, l'argument est considéré comme le nom d'une variable et c'est la valeur associée à cette variable qui est utilisée.

La plupart des fonctions SQL renvoient un code de résultat. Un code de résultat différent de zéro signale un échec de la fonction, et d'autres actions doivent alors être effectuées. Le code de résultat peut être utilisé par la fonction `SQLErrorMsg()`. L'insertion de fonctions SQL dans des scripts s'effectue dans l'éditeur de QuickScripts d'InTouch.

Conclusion :

Dans ce chapitre nous avons étudié l'automate S7-300 de la firme siemens, le logiciel KEPServer qui permet la connexion entre l'automate et InTouch, on a aussi étudié le système de contrôle et de supervision de données SCADA InTouch, ces caractéristique principales ainsi que son rôle dans le contrôle et la supervision de données, puis nous avons étudié les bases de donnée et les SGBD. Ensuite nous avons décrit les liaisons entre l'automate et chaque logiciel associer.

Chapitre III

Exemple de sauvegarde et teste requête

Introduction

Pour sauvegarder nos données dans une base de données nous allons réaliser des liaisons entre les différentes applications et automate cité auparavant.

Dans, ce chapitre nous allons décrire les liaisons et présenté un exemple de requête et de sauvegarde de données des variables identifiées à partir de l'analyse fonctionnelle, et terminé avec une conclusion.

III.1 Identification des paramètres de la zone 60 (DRINK) à sauvegarder sur la base de données

Tableau III.1 : variables de la zone 60 (DRINK) à sauvegarder.

Variables	Type de variable	unité	Adresse	Type API
60DT001	taux de Brix à la fin de la pasteurisation	°B	DB550.KG408	API_TADRINK
60TE044	température du produit chauffé	°C	DB550.KG480	API_TADRINK
60FT005	débit du produit à la sortie du dégazage	L/h	DB550.KG420	API_TADRINK
60TE026	température du produit refroidi	°C	DB550.KG460	API_TADRINK
60V008	pourcentage d'ouverture de la vanne modulante V08	%	DB550.KG636	API_TADRINK
60LE008	Niveau bac de lancement	%	DB550.KG432	API_TADRINK
60TE043A	Température du produit à l'entrée du dégazeur	°C	DB550.KG472	API_TADRINK
60TE043B	Température produit à la sortie du dégazeur	°C	DB550.KG476	API_TADRINK
60PT032A	Pression à la sortie du dégazeur	kPa	DB550.KG516	API_TADRINK
60PT032B	Dépression du dégazeur	kPa	DB550.KG520	API_TADRINK
60PT005	Pression dans le pasteurisateur	b	DB550.KG512	API_TADRINK
60TE045	Température de préchauffage	°C	DB550.KG484	API_TADRINK
60PT061A	Pression du produit avant refroidissement	kPa	DB550.KG524	API_TADRINK
60PT061B	Pression de l'eau chauffée	kPa	DB550.KG528	API_TADRINK
60PT063A	Pression du produit après refroidissement	kPa	DB550.KG540	API_TADRINK

60PT063B	Pression de l'eau glycolé	kPa	DB550.KG544	API_TADRINK
60Vol_RTD	Volume transféré vers de pasteurisateur	L/h	DB550.KG510	API_TADRINK

III.2 Langage de scripts InTouch

Le langage de script InTouch est utilisé pour automatiser des opérations répétitives dans une application. La boîte dialogue **Script Editor** (éditeur de scripts) de WindowMaker permet de créer et de modifier un script.

III.2.1 Fonctions SQL utilisé dans le Script

Parmi les fonctions utilisées dans notre script on cite :

III.2.1.1 Fonction SQLConnect

La fonction SQLConnect() s'utilise dans un QuickScript InTouch pour se connecter à une base de données spécifiée par l'argument *ConnectString*.<

SQLConnect() renvoie une valeur à l'argument *ConnectionID*, utilisé comme paramètre dans toutes les fonctions SQL suivantes. Une liste de liens doit avoir été définie dans le fichier SQL.DEF du dossier de l'application avant d'utiliser la fonction SQLConnect dans un script.

Syntax: [ResultCode=]SQLConnect(ConnectionID, "ConnectString");

III.2.1.2 Fonction SQLDisconnect

La fonction SQLDisconnect() effectue la déconnexion de la base de données et libère toutes les ressources obtenues au moyen des fonctions SQLPrepareStatement() et SQLInsertPrepare().

Syntax: [ResultCode=]SQLDisconnect(ConnectionID);

III.2.1.3 Fonction SQLExecute

La fonction SQLExecute() est utilisée dans un script pour exécuter une requête SQL. S'il s'agit d'une instruction SELECT, le paramètre *BindList* désigne le nom de la liste de liens permettant d'associer les colonnes de base de données aux variables InTouch. Si la liste de liens est nulle, aucune association de variables n'est effectuée.

Syntaxe: SQLExecute(*ConnectionID*,*BindList*,*StatementID*);

III.2.1.4 Fonction SQLErrorMsg

Toutes les fonctions SQL retournent un code de résultat pouvant servir à des fins de dépannage. La fonction SQLErrorMsg() retourne le message d'erreur associé à ce code sous forme de variable InTouch de type message.

Syntax: ErrorMsg=SQLErrorMsg(*ResultCode*);

III.2.1.5 Fonction SQLNext

La fonction SQLNext() sélectionne l'enregistrement suivant, dans la table de résultats créée par la dernière fonction SQLSelect(). Une fonction SQLSelect() doit être exécutée préalablement à l'utilisation de la fonction SQLNext().

Syntax: [ResultCode=]SQLNext(*ConnectionID*);

III.3.1.6 Fonction SQLPrepareStatement

La fonction SQLPrepareStatement() prépare l'exécution de l'instruction SQL. Cette fonction ne permet pas d'exécuter l'instruction ; elle se contente de l'activer pour que vous puissiez définir des valeurs pour les paramètres.

Syntax: SQLPrepareStatement(*ConnectionId*, *StatementID*);

III.3.1.7 Fonction SQLSetStatement

La fonction SQLSetStatement() démarre un tampon d'instructions SQL à l'aide du contenu de *SQLStatement*, sur la connexion établie, *ConnectionID*. Il ne peut y avoir qu'un seul tampon d'instructions SQL par *ConnectionID*. Les erreurs sont renvoyées par la fonction.

Syntax: [ResultCode=]SQLSetStatement(*ConnectionID*, *SQLStatement*);

III.3.1.8 Fonction SQLAppendStatement

La fonction SQLAppendStatement() poursuit une instruction SQL en utilisant le contenu d'une chaîne. Une valeur retour signale une erreur pendant l'appel de la fonction.

Les variables InTouch admettent des chaînes de 131 caractères maximum. La fonction SQLAppendStatement() s'utilise normalement pour concaténer de nouvelles chaînes à une instruction.

Syntaxe : [ResultCode=]SQLAppendStatement(*ConnectionID*, "*SQLStatement*");

III.3.1.9 Fonction SQLClearStatement

La fonction SQLClearStatement() libère les ressources de la base de données, associées à l'instruction identifiée par l'argument *StatementID*.

Syntax: [ResultCode=]SQLClearStatement(*ConnectionID*, *StatementID*);

III.3.1.10 Fonction StringFromReal

La fonction StringFromReal() permet de convertir dans un script une valeur de type réel en une valeur chaîne.

Si précisé, la fonction :

- Arrondit la valeur à la précision spécifiée.
- Fait passer la valeur à notation exponentielle.

Syntaxe : result =StringFromReal(*number*, *precision*, *type*);

III.4 Création d'une base de données

Après connexion au server souhaiter Une nouvelle fenêtre s'ouvre, elle représente l'interface initiale de l'outil.

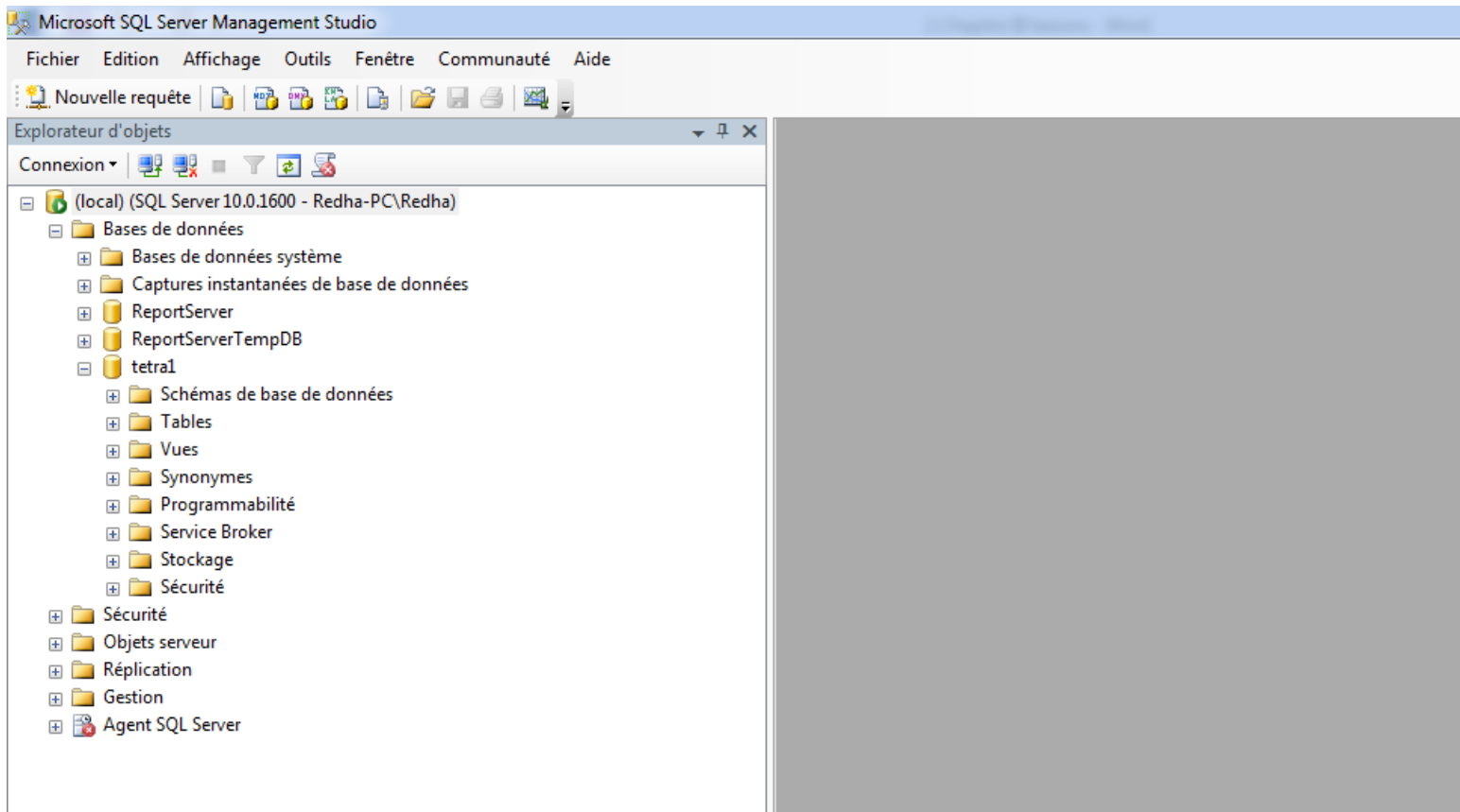


Figure III.1 : Interface Microsoft SQL Server Management Studio.

On voit que cette fenêtre est divisée en deux parties, d'une part la fenêtre « Explorer » à gauche qui nous affiche tous les objets du serveur et nous permet de naviguer entre ces différents composants ; la partie droite est une autre fenêtre qui nous montre les données et les informations sous forme de sommaire.

Pour créer notre base de données, on doit cliquer avec le bouton droit sur «Bases de données » dans la fenêtre « Explorer », puis on clique sur « Nouvelle base de données... ».

Dans la nouvelle fenêtre qui s'affiche, on va donner le nom «tetra1» à notre base puis on clique sur « Ok »

On peut maintenant voir, à partir de la fenêtre « Explorer », que notre base est créée.

III.5 Création des tables

Toujours dans la fenêtre « Explorer » on clique avec le bouton droit sur « Tables » dans notre base « tetra1 », puis on clique sur « New Table ».

À droite et à la même place de la fenêtre « Sommaire » une nouvelle fenêtre s'ouvre, dans laquelle on doit saisir toutes les colonnes de la table et le type de données. Après avoir saisi toutes ces données, on enregistre cette table en la nommant.

Nos deux tables (TETRA_DR1 et TETRA_DR2) sont maintenant créées. Pour vérifier que la création est bien réalisée on peut naviguer dans la fenêtre « Explorer » et voir si nos deux tables existent vraiment.

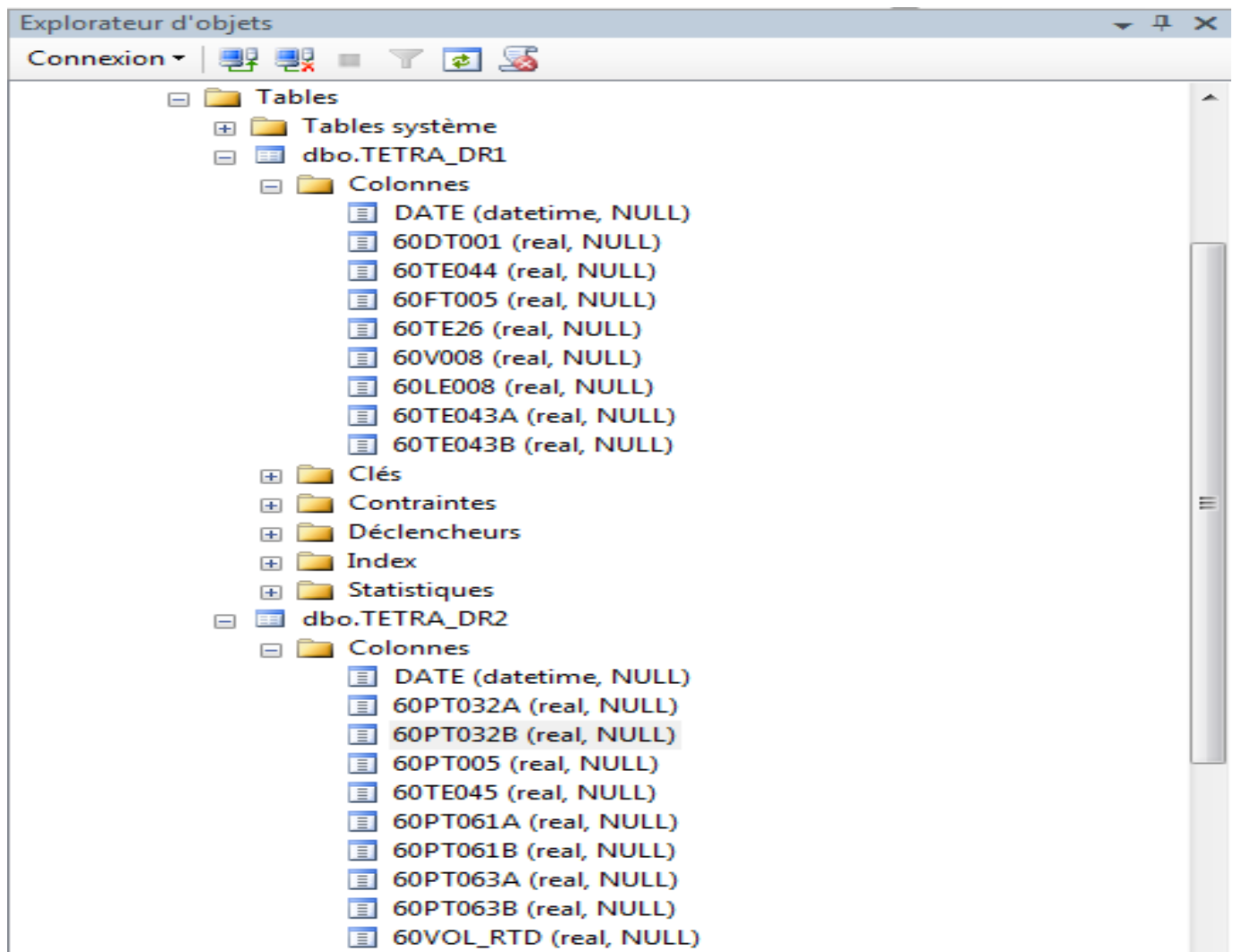


Figure III.2 : Fenêtre explorateur représentant les deux tables TETRA_DR1 et TETRA_DR2 créé avec les données des colonnes

III.6 Algorithme du script utilisé sur InTouch pour la sauvegarde de données dans tetra1

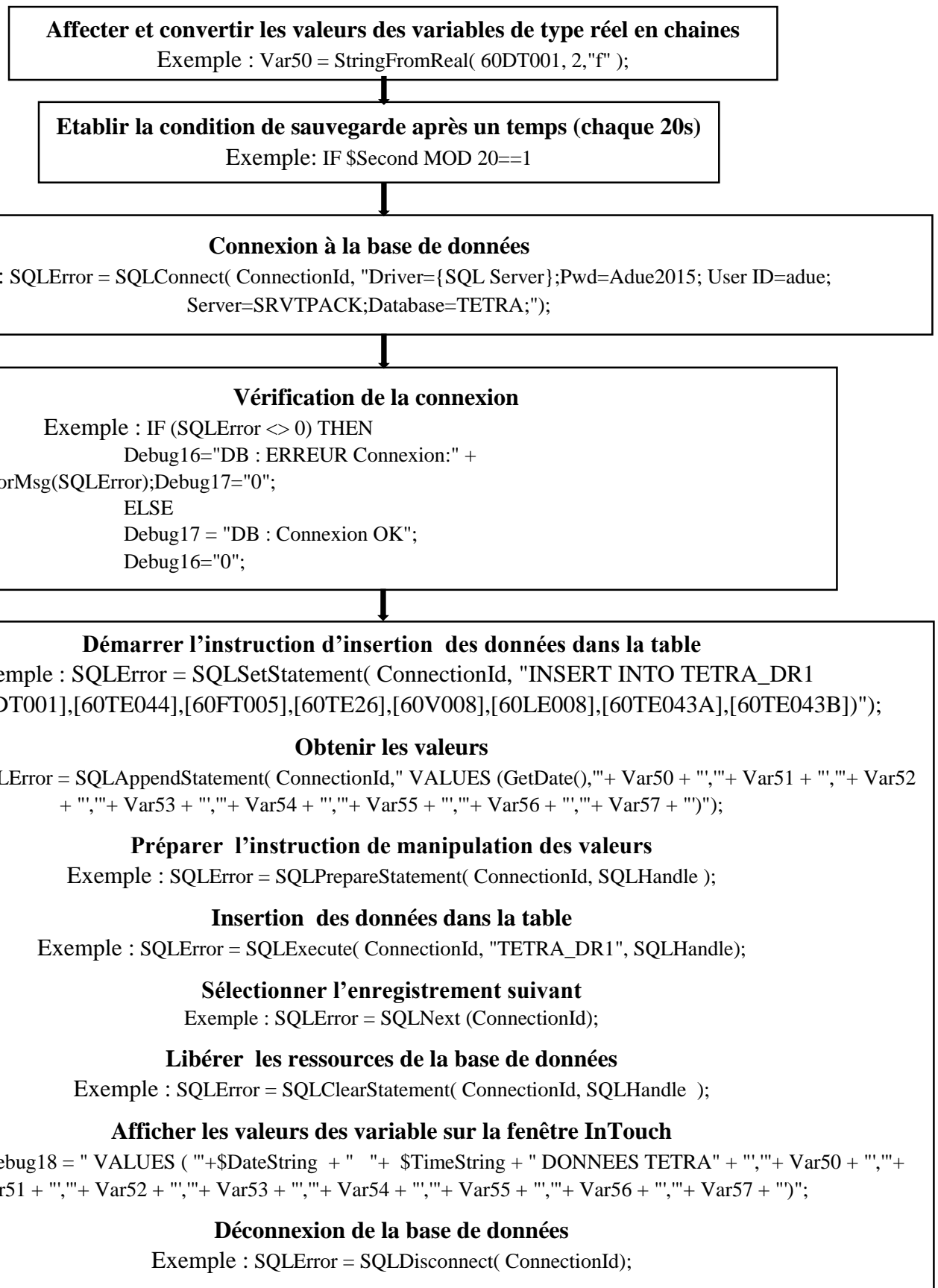


Figure III.3 : Algorithme du script de sauvegarde de données sur InTouch dans la base tetra1

III.7 Algorithme utilisée dans Sage 100 Edition Pilotée à fin de récupérer les valeurs de la base de données tetra1

Après connexion à la base tetra1 on procède à la récupération des données de chaque variable des deux tables TETRA_DR1 et TETRA_DR2 on utilisant deux scripts voir l'ANEX.

Sélectionner toutes les données des variables de la table TETRA_DR1.
Etablir deux Condition de sélection indiquez une date de début et une date de fin

Figure III.4 : Algorithme du script de récupération des données de la table TETRA_DR1

On procède de la même manière pour l'obtention des données de la table TETRA_DR2.

III.8 Test requête SQL Server :

On formule des requêtes on choisissant une date et une heure de début et de fin pour visualisé les données sauvegardées sur un tableau ou en forme de graphe. Cela à l'aide du script d'obtention des données de la base tetra1 'voir l'ANNEX' de l'application 'Sage 100 Edition Pilotée'.

Après avoir formulé quelques requêtes on aura les résultats sur les figures suivantes :

TableauxIII.2 : Tableau des données sauvegardées sur la table 1 de la base tetra1.

DATE(SQL 1 avé	60DT001	60TE044	60FT005	60TE26	60V008	60LE008	60TE043A	60TE043B
06/06/17 01:00	0,00	100,50	37847,22	20,02	100,00	98,83	54,05	51,39
06/06/17 01:00	0,00	100,54	37868,92	20,07	100,00	98,98	54,17	51,48
06/06/17 01:00	0,00	100,53	37919,56	20,05	100,00	98,83	54,07	51,50
06/06/17 01:01	0,00	100,53	37883,39	20,05	100,00	98,91	54,07	51,41
06/06/17 01:01	0,00	100,50	37774,88	20,02	100,00	98,98	54,07	51,44
06/06/17 01:01	0,00	99,76	37811,05	19,98	100,00	99,06	54,14	51,53
06/06/17 01:02	0,00	99,87	37760,42	19,93	100,00	98,98	54,19	51,53
06/06/17 01:02	0,00	100,13	37586,80	19,91	100,00	99,36	54,10	51,46
06/06/17 01:02	0,00	99,95	37897,86	19,95	100,00	98,98	54,14	51,55
06/06/17 01:03	0,00	99,94	37868,92	20,02	100,00	98,91	54,17	51,46
06/06/17 01:03	0,00	99,94	37760,42	20,02	100,00	98,91	54,17	51,53
06/06/17 01:03	0,00	99,90	37745,95	20,05	100,00	98,68	54,14	51,60
06/06/17 01:04	0,00	99,86	37796,59	20,05	100,00	98,83	54,31	51,57
06/06/17 01:04	0,00	99,80	37680,84	20,02	100,00	99,28	54,14	51,46
06/06/17 01:04	0,00	99,69	37854,46	20,02	100,00	99,36	54,14	51,46
06/06/17 01:05	0,00	99,66	37782,12	20,02	100,00	99,06	54,31	51,53
06/06/17 01:05	0,00	99,65	37782,12	20,00	100,00	98,91	54,21	51,48
06/06/17 01:05	0,00	99,65	37622,97	20,00	100,00	99,06	54,14	51,41

Tableau III.3: Tableau des données sauvegardées sur la table 2 de la base tetral.

DATE(SQL 2 av	60PT005	60PT061A	60PT061B	60PT063A	60PT063B	60TE032A	60TE032B	60TE045	60VOL_RTD
11/06/17 12:15	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:15	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:15	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:16	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:16	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:16	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:17	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:17	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:18	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:19	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:19	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:19	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 12:20	171,44	735,53	496,67	563,51	227,43	-77,10	-84,62	95,20	18 830,00
11/06/17 02:43	298,61	796,59	496,67	621,38	160,88	-79,05	-86,97	99,69	8 010,00
11/06/17 02:43	300,35	803,96	495,95	635,85	163,05	-79,51	-86,99	99,75	8 010,00
11/06/17 02:43	295,28	787,18	495,52	649,59	164,64	-80,56	-86,93	99,75	8 010,00

Remarque : On remarque que les valeurs des données dans les deux tableaux s’affiche tous et avec le temps préciser.

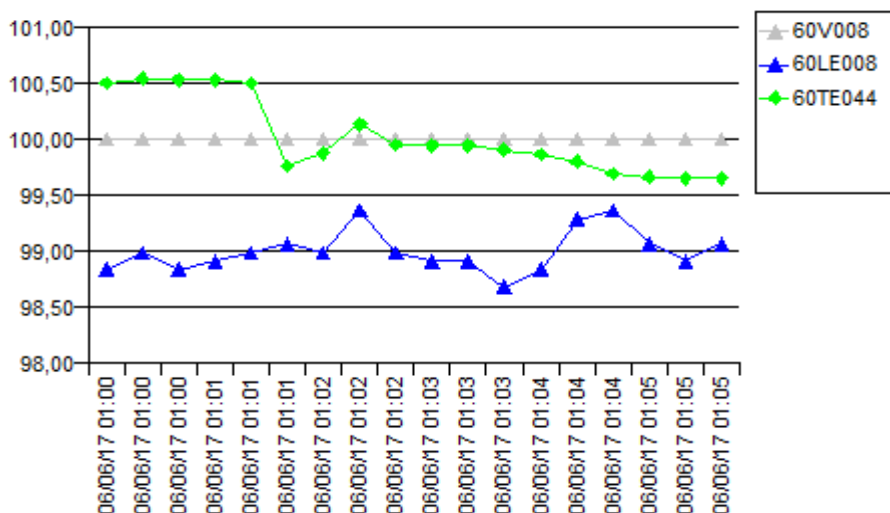


Figure III.5 : Courbe de données de trois variables d’une requête formulée.

Remarque : on voit dans cette courbe l’évolution de température du produit chauffé et le pourcentage d’ouverture de la vanne modulante V08 et le niveau bac de lancement en fonction du temps

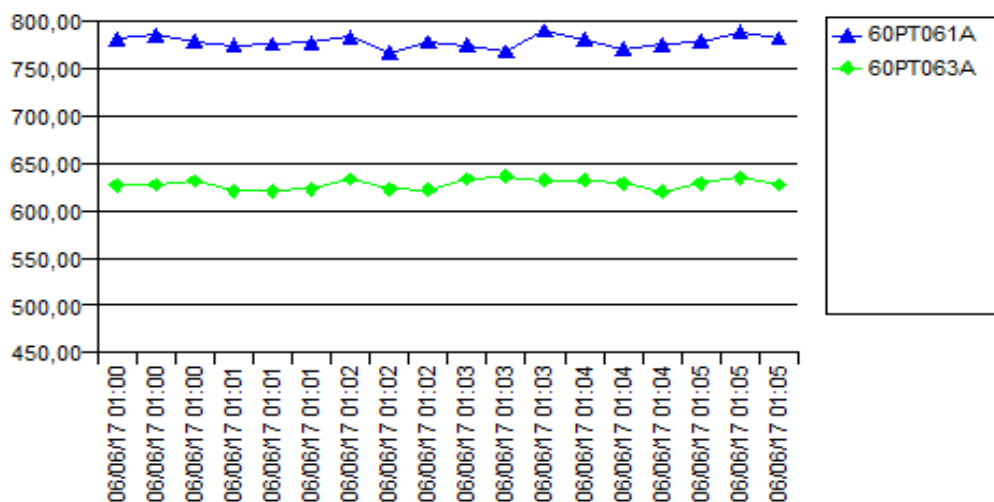


Figure III.6 : Courbe de données de deux variables d’une requête formulée.

Remarque : on voit dans cette courbe l’évolution de la pression du produit avant et après refroidissement en fonction du temps

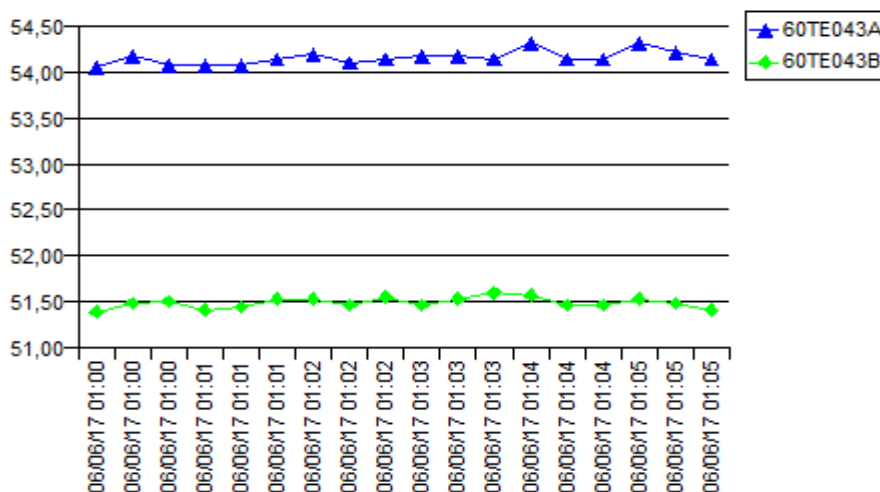


Figure III.7: Courbe de données de deux variables d’une requête formulée.

Remarque : on voit dans cette courbe l’évolution de la température à l’entrée et à la sortie du dégazeur en fonction du temps.

Conclusion

Au cours de ce chapitre, nous avons présenté les paramètres du système à sauvegarder. Ensuite nous avons décrit un algorithme qui nous permettra la sauvegarde des données de ces paramètres. Puis nous avons présenté les différentes étapes de la création d’une base de données et d’une table. Enfin, nous avons présenté quelques requêtes et visualisé sous formes de tableau et de graphe le comportement de ces paramètres.

CONCLUSION GENERALE

Notre travail s'est porté sur l'élaboration d'un historique des paramètres du système de production, et ce en utilisant l'automate S7-300, le logiciel de supervision InTouch, le système de communication KEPServer et Microsoft SQL Server pour les bases de données.

Pour atteindre notre objectif, nous avons commencé par prendre connaissance du système de production, étudié le fonctionnement du processus puis identifié les différents équipements et instruments qui le constituent.

Afin d'élaborer l'historique de ces paramètres, l'étude et l'élaboration de l'analyse fonctionnelle a été effectuée.

Le passage en revue des automates programmables industriels en particulier l'automate S7-300, ainsi que les trois logiciels utilisés à savoir InTouch, KEPServer et Microsoft SQL Server, leurs caractéristiques ainsi que les différentes liaisons entre ces trois logiciels et l'automate ont été abordés.

L'élaboration de l'analyse fonctionnelle qui a été effectuée et l'utilisation de l'automate S7-300 et la prise en connaissance de ces trois logiciels nous a permis de récupérer les données des paramètres production du système permettant l'amélioration des ensembles des processus de production, et la garanti de la qualité des produits et des services.

Le travail que nous avons mené au sein de la "SARL IBRAHIM & fils, IFRI" nous a permis d'acquérir les connaissances techniques et pratiques qui viennent compléter les enseignements théoriques acquises, et nous offre la possibilité de familiariser avec le milieu industriel et avec la réalité du terrain dans lequel nous sommes appelés à travailler.

Comme perspective nous espérons que les autres services de l'entreprise seront doté de systèmes de gestion d'historique pour de répondre aux défauts et les pannes à venir à fin de faire avancer la productivité et améliorer les conditions de travail.

Enfin, nous espérons que notre travail puisse servir comme support pour des solutions archivages.

Bibliographie

- [1] N.Kandi, «automate programmable industriel », document de formation LAP Boumerdes.
- [2] Alain Gonnzaga, «les automates programmables industriels».
- [3] Manuel Siemens, logiciel STEP7, 2004.
- [4] Philippe Brun, «automate programmable », lycée luis ARNAUD, Strasbourg édition décembre.
- [5] Solutions de réseaux pour PROFIBUS conformes à CEI 61158 et EN 50170 Brochure technique Avril 2008.
- [6] « Supervision SCADA », Actors Solutions (consulté le 14 octobre 2013).
- [7] OFFICE OF THE MANAGER NATIONAL COMMUNICATIONS SYSTEM, « Supervisory Control and Data Acquisition (SCADA) Systems », NATIONAL COMMUNICATIONS SYSTEM, octobre 2004.
- [8] « The new generation of technology Scada / HMI », Progea Srl, 2013.
- [9] « Un an d'informatique industrielle », Jautomatise N° 44 - Janvier-Février 2006.
- [10] « CONCEPTION D'UNE INTERFACE DE SUPERVISION », Thomas GHIEN 2004-2005.
- [11] Concepts et fonctionnalités d'InTouch, ©2007 Invensys Systems, Inc. Tous droits réservés.
- [12] L'Isambert, ''Pneumatique dans les réseaux industriels''.
- [13] Michel Grout et Patrick Salaun, Instrumentation industriel : 3^{ème} Edition.
- [14] A.ROMON, M.TOLED0.Introduction aux bases de données relationnelles. Shaums. 2002.
- [15] Bases de données de la modélisation au SQL Laurent AUDIBERT.
- [16] SQL SERVER 2008- SQL, Transact SQL Conception et réalisation d'une base de données.
- [17] Help copyright 1996-2009 Kepware Technologies.

A. Script de Sauvegarde des paramètres de la Z60 sur la Base de données Tétra

```

Var50 = StringFromReal( 60DT001, 2,"f" );
Var51 = StringFromReal( 60TE044, 2,"f" );
Var52 = StringFromReal( 60FT005, 2,"f" );
Var53 = StringFromReal( 60TE026, 2,"f" );
Var54 = StringFromReal( 60V008, 2,"f" );
Var55 = StringFromReal( 60LE008, 2,"f" );
Var56 = StringFromReal( 60TE043A, 2,"f" );
Var57 = StringFromReal( 60TE043B, 2,"f" );
Var58 = StringFromReal( 60PT032A,2, "f" );
Var59 = StringFromReal( 60PT032B,2, "f" );
Var60 = StringFromReal( 60PT005,2, "f" );
Var61 = StringFromReal( 60TE045,2, "f" );
Var62 = StringFromReal( 60PT061A,2, "f" );
Var63 = StringFromReal( 60PT061B, 2,"f" );
Var64 = StringFromReal( 60PT063A, 2,"f" );
Var65 = StringFromReal( 60PT063B, 2,"f" );
Var66 = StringFromReal( VOL_RTD, 2,"f" );

```

```
IF $Second MOD 20 == 1 THEN
```

```

Debug16= "0";
Debug17 = "0";
Debug18= "0";
Debug19= "0";

```

```
{ Connexion à la base de données SQL Serveur }
```

```

SQLError = SQLConnect( ConnectionId, "Driver={SQL Server};Pwd=Adue2015; User
ID=adue; Server=SRVTPACK;Database=TETRA;");

```

```
IF (SQLError <> 0) THEN
```

```

Debug16="DB : ERREUR Connexion:" + SQLErrorMsg(SQLError);
Debug17="0";

```

```
ELSE
```

```

Debug17 = "DB : Connexion OK";
Debug16="0";

```

```

SQLError = SQLSetStatement( ConnectionId, "INSERT INTO TETRA_DR1
(Date,[60DT001],[60TE044],[60FT005],[60TE26],[60V008],[60LE008],[60TE043A],[60TE0
43B])");

```

```

SQLError = SQLAppendStatement( ConnectionId," VALUES (GetDate(),"+ Var50 +
", "+ Var51 + ", "+ Var52 + ", "+ Var53 + ", "+ Var54 + ", "+ Var55 + ", "+ Var56 + ", "+
Var57 + ")");

```

```
SQLError = SQLPrepareStatement( ConnectionId, SQLHandle );
```

```
SQLError = SQLExecute( ConnectionId, "TETRA_DR1", SQLHandle);
```



```

SQLError = SQLNext (ConnectionId);
SQLError = SQLClearStatement( ConnectionId, SQLHandle );

Debug18 = " VALUES ( '"+$DateString + " " + $TimeString + " DONNEES TETRA" +
"',"+ Var50 + "',"+ Var51 + "',"+ Var52 + "',"+ Var53 + "',"+ Var54 + "',"+ Var55 + "',"+
Var56 + "',"+ Var57 + "')";

SQLError = SQLSetStatement( ConnectionId, "INSERT INTO TETRA_DR2
(Date,[60TE032A],[60TE032B],[60PT005],[60TE045],[60PT061A],[60PT061B],[60PT063A
],[60PT063B],[60VOL_RTD])");
SQLError = SQLAppendStatement( ConnectionId," VALUES (getDate(),"+ Var58 +
"',"+ Var59 + "',"+ Var60 + "',"+ Var61 + "',"+ Var62 + "',"+ Var63 + "',"+ Var64 + "',"+
Var65 + "',"+ Var66 + "')");
SQLError = SQLPrepareStatement( ConnectionId, SQLHandle );
SQLError = SQLExecute( ConnectionId, "TETRA_DR2", SQLHandle);
SQLError = SQLNext (ConnectionId);
SQLError = SQLClearStatement( ConnectionId, SQLHandle );

Debug19 = " VALUES ( '"+$DateString + " " + $TimeString + " DONNEES TETRA" +
"',"+ Var58 + "',"+ Var59 + "',"+ Var60 + "',"+ Var61 + "',"+ Var62 + "',"+ Var63 + "',"+
Var64 + "',"+ Var65 + "',"+ Var66 + "')";

ENDIF;

SQLError = SQLDisconnect( ConnectionId);

ENDIF;

```

B. Script utilisée dans Sage 100 Edition Pilotée à fin d’obtenir les valeurs de la base de données tetra1

Script d’obtention des données de la table TETRA_DR1 :

```

SELECT
    [DATE]
    ,[60DT001]
    ,[60TE044]
    ,[60FT005]
    ,[60TE26]
    ,[60V008]
    ,[60LE008]
    ,[60TE043A]
    ,[60TE043B]

FROM [tetra1].[dbo].[TETRA_DR1]
where
([DATE]>= @Prompt ('Indiquez une date de début ','A',MONO,FREE)
and
[DATE]<= @Prompt ('Indiquez une date de fin ','A',MONO,FREE) )

```

Script d'obtention des données de la table TETRA_DR2 :

```
SELECT
    [DATE]
    ,[60TE032A]
    ,[60TE032B]
    ,[60PT005]
    ,[60TE045]
    ,[60PT061A]
    ,[60PT061B]
    ,[60PT063A]
    ,[60PT063B]
    ,[60VOL_RTD]
FROM [tetra1].[dbo].[TETRA_DR2]
where
([DATE]>= @Prompt ('Indiquez une date de début ','A',,MONO,FREE)
and
[DATE]<= @Prompt ('Indiquez une date de fin ','A',,MONO,FREE) )
```


Local Compresseur
Alsafe

REV. C	DPI	Modification suite changement de site	16/11/9	DPI
REV. VRS				
SCHEMA Implantation		DESCRIPTION		
IFRI ALGERIE		Tetra Pak SERVICE SA		
PROJ. No	235-901	DATE	9/9/9	PLAN No
NOI/PROJ	REV. No			
ARCHIVE	ACAD			

Ca document est la propriété de Tetra Pak & Cie. Logotype, il ne peut être réutilisé sans son autorisation écrite. Révisé et corrigé en conséquence. 3ème édition. Version précédente: 2ème édition. Plan: 01/01/99. 15/11/99.

